IEICE TRANS. INF. & SYST., VOL.E94–D, NO.2 FEBRUARY 2011

PAPER Special Section on Foundations of Computer Science — Mathematical Foundations and Applications of Algorithms and Computer Science —

Improved Approximation Algorithms for Firefighter Problem on Trees

Yutaka IWAIKAWA[†a)], **Naoyuki KAMIYAMA**[†b)], ***Nonmembers, and*** **Tomomi MATSUI**[†c)], ***Member***

SUMMARY The firefighter problem is used to model the spread of fire, infectious diseases, and computer viruses. This paper deals with firefighter problem on rooted trees. It is known that the firefighter problem is $\mathcal{NP}$-hard even for rooted trees of maximum degree 3. We propose techniques to improve a given approximation algorithm. First, we introduce an implicit enumeration technique. By applying the technique to existing $(1-\frac{1}{e})$-approximation algorithm, we obtain $(1-\frac{k-1}{(k-1)e+1})$-approximation algorithm when a root has k children. In case of ternary trees, $k=3$ and thus the approximation ratio satisfies $(1-\frac{k-1}{(k-1)e+1}) \geq 0.6892$, which improves the existing result $1-\frac{1}{e} \geq 0.6321$. Second technique is based on backward induction and improves an approximation algorithm for firefighter problem on ternary trees. If we apply the technique to existing $(1-\frac{1}{e})$-approximation algorithm, we obtain 0.6976-approximation algorithm. Lastly, we combine the above two techniques and obtain 0.7144-approximation algorithm for firefighter problem on ternary trees.
key words: *firefighter problem, approximation algorithm, rooted tree*

1. Introduction

The *firefighter problem* was introduced by Hartnell [4] and can be used to model the spread of fire, infectious diseases, and computer viruses. The firefighter problem is defined as follows. We are given a graph, a specified vertex called *root*, and nonnegative vertex weights. At time 0, a fire breaks out at the root. At each subsequent time step, a firefighter deploys a vertex which is not yet on fire and defends it, and then the fire spreads to all unprotected adjacent vertices of each burned vertex. The process ends when the fire can no longer spread, and all the vertices which are not burning are considered saved. The objective of the firefighter problem is to determine posture of firefighters so as to maximize the sum of weights of saved vertices.

This paper deals with the firefighter problem on a rooted tree. We propose two techniques to improve a given approximation algorithm. First, we introduce an implicit enumeration technique. By applying the technique to existing α-approximation algorithm, we obtain $(1-\frac{(k-1)(1-\alpha)}{(k-1)+(1-\alpha)})$-approximation algorithm when the root has k children. If we employ $(1-\frac{1}{e})$-approximation algorithm proposed in [2], the technique gives $(1-\frac{k-1}{(k-1)e+1})$-approximation algorithm. Second technique is based on backward induction and improves an approximation algorithm for firefighter problem on ternary trees. If we apply the technique to existing α-approximation algorithm, an approximation ratio of obtained algorithm is greater than or equal to $(\alpha+\sqrt{(1-\alpha)^2+1}-1)$. By combining the above two techniques, we propose 0.7144-approximation algorithm for firefighter problem on ternary trees.

It is known [3] that the firefighter problem is $\mathcal{NP}$-hard even for rooted trees of maximum degree 3. Hartnell and Li [5] have proved that a simple greedy method gives a 0.5-approximation algorithm. Cai, Verbin and Yang [2] proposed a randomized $(1-\frac{1}{e})$-approximation algorithm (note $1-\frac{1}{e} \approx 0.6321$). Recently, Anshelevich, Chakrabarty, Hate and Swamy [1] showed that this result can be also derived from a reduction to the submodular function maximization problem.

2. Implicit Enumeration Technique

Let T be a given rooted tree and $w(v)$ be a vertex weight of vertex v in T. For any subtree T' of the given tree T, we introduce the following notations. The vertex set of T' is also denoted by T' when there is no ambiguity. We denote a firefighter problem defined on T' by $\mathrm{FF}(T')$ where a weight of a vertex in T' is equal to that in T. For any vertex v of T, a *subtree rooted at* v is a subtree of T induced by set of descendants of v (including v). Given a vertex subset T' of T, we use the notation $w(T') = \sum_{v \in T'} w(v)$.

In this section, we propose an approximation algorithm based on implicit enumeration technique. It is easy to see that every firefighter problem has an optimal solution satisfying that we put a firefighter to a child of the root. Thus, we can improve a solution by considering all the cases that exactly one child of the root has a firefighter. In our first algorithm, we solve (small) firefighter problems by a given α-approximation algorithm, where $0 \leq \alpha \leq 1$.

Algorithm IE (α)

Step 1: Let C be a set of children of the root of T. For each vertex $i \in C$, T_i denotes a subtree of T rooted at i.

Step 2: For each vertex $i \in C$, we execute the following.

1. Construct a tree $\overline{T}_i$ by contracting the roots of $|C|-1$ trees in $\{T_j \mid j \in C\} \setminus \{T_i\}$ to a single vertex (see Fig. 1).
2. Apply an α-approximation algorithm to problem $\mathrm{FF}(\overline{T}_i)$. We denote an obtained solution (set of vertices with firefighters) by $\overline{F}_i$ and set of saved vertices by $\overline{S}_i$.

Manuscript received March 29, 2010.

†The authors are with the Department of Information and System Engineering, Faculty of Science and Engineering, Chuo University, Tokyo, 112–8551 Japan.

a) E-mail: fss-5stars@hotmail.co.jp

b) E-mail: kamiyama@ise.chuo-u.ac.jp

c) E-mail: matsui@ise.chuo-u.ac.jp

DOI: 10.1587/transinf.E94.D.196

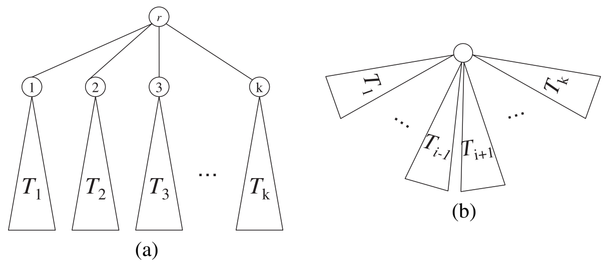

Fig. 1 (a) Subtree T_i. (b) Tree $\overline{T}_i$.

Step 3: Find $i^* \in C$ which attains $\max\{w(T_j \cup \overline{S}_j) \mid j \in C\}$. Output set $\{i^*\} \cup \overline{F}_{i^*}$ of vertices with firefighters and set of saved vertices $T_{i^*} \cup \overline{S}_{i^*}$.

Let us discuss the approximation ratio of Algorithm IE(α). In the following, we assume that the root has k children, i.e., $|C| = k$. For any subtree T', we denote the optimal value of FF(T') by $z(T')$.

Lemma 1: For any vertex $j \in C$, we have the followings.

(1) $\displaystyle\sum_{i \in C \setminus \{j\}} w(T_i) \geq z(\overline{T}_j)$,

(2) $\displaystyle\sum_{i \in C \setminus \{j\}} z(\overline{T}_i) \geq (k-2)\, z(\overline{T}_j)$.

Proof. (1) A set of saved vertices by an optimal solution of problem FF($\overline{T}_j$) is a subset of non-root vertices in $\overline{T}_j$. A set of non-root vertices of $\overline{T}_j$ is a subset of $\cup_{i \in C \setminus \{j\}} T_i$, and thus the inequality is obvious.
(2) Let $\overline{S}^*_j$ be a set of saved vertices by an optimal solution (set of vertices with firefighters) $\overline{F}^*_j$ of problem FF($\overline{T}_j$). For any $i \in C \setminus \{j\}$, vertex set $\overline{F}^*_j \setminus T_i$ is feasible to problem FF($\overline{T}_i$) and corresponding objective value is equal to $w(\overline{S}^*_j \setminus T_i)$. Thus, inequality $z(\overline{T}_i) \geq w(\overline{S}^*_j \setminus T_i)$ holds and accordingly we have that

$$\begin{aligned}
\sum_{i \in C \setminus \{j\}} z(\overline{T}_i) &\geq \sum_{i \in C \setminus \{j\}} w(\overline{S}^*_j \setminus T_i) \\
&= \sum_{i \in C \setminus \{j\}} \left(w(\overline{S}^*_j) - w(\overline{S}^*_j \cap T_i) \right) \\
&= (k-1) w(\overline{S}^*_j) - \sum_{i \in C \setminus \{j\}} w(\overline{S}^*_j \cap T_i) \\
&= (k-1) w(\overline{S}^*_j) - w(\overline{S}^*_j) \\
&= (k-2) w(\overline{S}^*_j) = (k-2) z(\overline{T}_j).
\end{aligned}$$

This completes the proof. □

We are now ready to prove the main theorem in this section.

Theorem 1: Algorithm IE(α) is $(1 - \frac{(k-1)(1-\alpha)}{(k-1)+(1-\alpha)})$-approximation algorithm.

Proof. Due to the definition of $\overline{S}_i$ obtained in Step 2, inequality $w(\overline{S}_i) \geq \alpha z(\overline{T}_i)$ holds for any $i \in C$. For any vertex $j \in C$ and θ such that $0 \leq \theta \leq 1$, we have that

$$\begin{aligned}
& w(T_{i^*}) + w(\overline{S}_{i^*}) \\
&\quad = \max_{i \in C} (w(T_i) + w(\overline{S}_i)) \\
&\quad \geq (1-\theta) \left(w(T_j) + w(\overline{S}_j) \right) \\
&\qquad + \sum_{i \in C \setminus \{j\}} \frac{\theta}{k-1} \left(w(T_i) + w(\overline{S}_i) \right) \\
&\quad = (1-\theta) \left(w(T_j) + w(\overline{S}_j) \right) \\
&\qquad + \frac{\theta}{k-1} \left(\sum_{i \in C \setminus \{j\}} w(T_i) + \sum_{i \in C \setminus \{j\}} w(\overline{S}_i) \right) \\
&\quad \geq (1-\theta) \left(w(T_j) + \alpha z(\overline{T}_j) \right) \\
&\qquad + \frac{\theta}{k-1} \left(\sum_{i \in C \setminus \{j\}} w(T_i) + \sum_{i \in C \setminus \{j\}} \alpha z(\overline{T}_i) \right) \\
&\quad \geq (1-\theta) \left(w(T_j) + \alpha z(\overline{T}_j) \right) \\
&\qquad + \frac{\theta}{k-1} \left(z(\overline{T}_j) + \alpha(k-2) z(\overline{T}_j) \right) \\
&\quad = (1-\theta) \left(w(T_j) + \alpha z(\overline{T}_j) \right) \\
&\qquad + \theta \, \frac{1 + \alpha(k-2)}{k-1} z(\overline{T}_j).
\end{aligned} \tag{1}$$

By setting $\theta' = \frac{(k-1)(1-\alpha)}{(k-\alpha)}$, following inequality and equality

$$\begin{aligned}
w(T_{i^*}) + w(\overline{S}_{i^*}) &\geq (1-\theta') \left(w(T_j) + \alpha z(\overline{T}_j) \right) \\
&\qquad + \theta' \, \frac{1 + \alpha(k-2)}{k-1} z(\overline{T}_j) \\
&= \left(1 - \frac{(k-1)(1-\alpha)}{k-\alpha} \right) (w(T_j) + z(\overline{T}_j))
\end{aligned}$$

hold for any $j \in C$. Since the optimal value $z(T)$ of original problem FF(T) satisfies $z(T) = \max_{j \in C}(w(T_j) + z(\overline{T}_j))$, we have the desired result that

$$\begin{aligned}
& w(T_{i^*}) + w(\overline{S}_{i^*}) \\
&\quad \geq \left(1 - \frac{(k-1)(1-\alpha)}{(k-1)+(1-\alpha)} \right) \max_{j \in C} (w(T_j) + z(\overline{T}_j)) \\
&\quad = \left(1 - \frac{(k-1)(1-\alpha)}{(k-1)+(1-\alpha)} \right) z(T).
\end{aligned}$$

This completes the proof. □

By applying the above to $(1 - \frac{1}{e})$-approximation algorithm proposed in [2], we have the following result easily.

Corollary 1: If we employ $(1 - \frac{1}{e})$-approximation algorithm, the approximation ratio of algorithm IE($1 - \frac{1}{e}$) becomes $1 - \frac{k-1}{(k-1)e+1}$ when the root has k children.

In the rest of this section, we deal with k-ary trees. If we apply algorithm IE recursively, we can improve the approximation ratio. Let A_0 be $(1 - \frac{1}{e})$-approximation algorithm proposed in [2], and A_m be algorithm IE($\cdot$) which employ Algorithm A_{m-1} at Step 2. If we apply algorithm A_m

Table 1 Approximation ratios of Algorithms $A_0, \dots, A_3$ for k-ary Tree.

algorithm	ternary tree	4-ary tree	5-ary tree
A_0	0.6321205	0.6321205	0.6321205
A_1	0.6892751	0.6723046	0.6631047
A_2	0.7074553	0.6817844	0.6689742
A_3	0.7134432	0.6841220	0.6701359

to k-ary trees, the total number of execution of algorithm A_0 is $k^{O(m^2)}$. Table 1 shows approximation ratios of Algorithms $A_0, \dots, A_3$ when we applied to k-ary trees where $k \in \{3, 4, 5\}$.

3. Backward Induction Technique for Ternary Trees

In this section, we propose an approximation algorithm based on backward induction for ternary trees. Throughout this section, we assume that a given original tree T is ternary. We assume that the vertices of T are indexed by $\{1, 2, \dots, n\}$ satisfying that if vertex i is a child of vertex j, then $i < j$. Obviously, n is the root of T.

Algorithm BI (α)

For $i = 1$ to n, do {
  If vertex i is a leaf of T, then set $\widehat{z}(i) := 0$, $F_i := \emptyset$,
    and $S_i := \emptyset$.
  Else {
    Let T' be a subtree rooted at i.
    Apply algorithm IE(α) to FF(T') and obtain
      a solution F' and set of saved vertices S'.
    Find an ordered pair of distinct children (u^*, v^*)
      of i which maximizes the value $w(T_{u^*}) + \widehat{z}(v^*)$
      where T_{u^*} denotes a vertex set of a subtree
      rooted at u^*.
    If $w(S') > w(T_{u^*}) + \widehat{z}(v^*)$, then set $\widehat{z}(i) = w(S')$,
      $F_i = F'$ and $S_i = S'$.
    Else, set $\widehat{z}(i) = w(T_{u^*}) + \widehat{z}(v^*)$, $F_i = \{u^*\} \cup F_{v^*}$ and
      $S_i = T_{u^*} \cup S_{v^*}$.
  }
}

Let us discuss the approximation ratio of our algorithm.

Theorem 2: Algorithm BI(α) is $(\alpha + \sqrt{(1-\alpha)^2 + 1} - 1)$-approximation algorithm when a given tree is ternary.

Proof. In the following, we abbreviate $\alpha + \sqrt{(1-\alpha)^2 + 1} - 1$ by β for simplicity. From the description of the algorithm, it is obvious that for any vertex v of T, the equality $\widehat{z}(v) = w(S_v)$ holds. We show that $\widehat{z}(v) \geq \beta z(T_v)$ for any vertex v of T by induction on the index of vertices, where $z(T_v)$ denotes the optimal value of FF(T_v).

When $i = 1$, vertex i is a leaf of original tree T and thus $0 = \widehat{z}(i) \geq \beta z(T_i) = 0$, where $S_1 = \emptyset$ and T_1 includes only one vertex.

Assuming that the inequality $\widehat{z}(v) \geq \beta z(T_v)$ holds for any vertex $v \in \{1, 2, \dots, i-1\}$, we consider vertex i. If i is a leaf vertex, we can show the inequality easily. Consider the case that i has three children $\{f, g, h\}$. Without loss of generality, we can assume that there exists an optimal solution of FF(T_i) which puts a firefighter on vertex f. Let $\overline{T}_f$ be a tree obtained by merging roots of T_g and T_h. Then $z(T_i) = w(T_f) + z(\overline{T}_f)$.

Since algorithm BI(α) executes IE(α), an inequality $\widehat{z}(i) \geq w(T_f) + \alpha z(\overline{T}_f)$ holds. The backward induction step and induction hypothesis imply that

$$\begin{aligned}\widehat{z}(i) &\geq \max\{w(T_g) + \widehat{z}(h), w(T_h) + \widehat{z}(g)\}\\ &\geq \max\{w(T_g) + \beta z(T_h), w(T_h) + \beta z(T_g)\}.\end{aligned}$$

Clearly from the definition of $\overline{T}_f$, $z(T_g) + z(T_h) \geq z(\overline{T}_f)$ holds.

From the above, for any real θ satisfying $0 \leq \theta \leq 1$, we have that

$$\begin{aligned}\widehat{z}(i) &\geq \max\{w(T_f) + \alpha z(\overline{T}_f),\\ &\qquad w(T_g) + \beta z(T_h), w(T_h) + \beta z(T_g)\}\\ &\geq (1-\theta)(w(T_f) + \alpha z(\overline{T}_f))\\ &\quad + \frac{\theta}{2}(w(T_g) + \beta z(T_h) + w(T_h) + \beta z(T_g))\\ &\geq (1-\theta)(w(T_f) + \alpha z(\overline{T}_f))\\ &\quad + \frac{\theta}{2}(z(T_g) + \beta z(T_h) + z(T_h) + \beta z(T_g))\\ &= (1-\theta)(w(T_f) + \alpha z(\overline{T}_f))\\ &\quad + \frac{\theta}{2}(1+\beta)(z(T_g) + z(T_h))\\ &\geq (1-\theta)(w(T_f) + \alpha z(\overline{T}_f)) + \frac{\theta}{2}(1+\beta) z(\overline{T}_f).\end{aligned}$$

By setting $\theta' = 1 - \beta$, it is easy to show that $0 \leq \theta' \leq 1$ and

$$\begin{aligned}\widehat{z}(i) &\geq (1-\theta')(w(T_f) + \alpha z(\overline{T}_f)) + \theta' \frac{1+\beta}{2} z(\overline{T}_f)\\ &= (1-\theta') w(T_f) + \frac{2(1-\theta')\alpha + \theta'(1+\beta)}{2} z(\overline{T}_f)\\ &= \beta w(T_f) + \frac{2\alpha\beta + 1 - \beta^2}{2} z(\overline{T}_f)\\ &= \beta w(T_f) + \frac{(\beta - 2\alpha + 2)(-\beta) + 2\beta + 1}{2} z(\overline{T}_f)\\ &= \beta w(T_f)\\ &\quad + \frac{(1-\alpha)^2 - (\beta - \alpha + 1)^2 + 2\beta + 1}{2} z(\overline{T}_f)\\ &= \beta w(T_f)\\ &\quad + \frac{(1-\alpha)^2 - ((1-\alpha)^2 + 1) + 2\beta + 1}{2} z(\overline{T}_f)\\ &= \beta(w(T_f) + z(\overline{T}_f)) = \beta z(T_i).\end{aligned}$$

Thus, we have a desired inequality $\widehat{z}(i) \geq \beta z(T_i)$. □

When we employ $(1 - \frac{1}{e})$-approximation algorithm proposed in [2], and then we obtain the algorithm BI($1 - \frac{1}{e}$), whose approximation ratio becomes

$$\sqrt{1 + \left(\frac{1}{e}\right)^2} - \frac{1}{e} \geq 0.6976416.$$

Lastly, we consider the case that we solve FF(T') in algorithm BI($\cdot$) by algorithm A_1, i.e., algorithm IE($1-\frac{1}{e}$) proposed in the previous section. Corollary 1 says that the approximation ratio of IE($1-\frac{1}{e}$) is ($1-\frac{5}{5e+1}$), since the root of $\overline{T}_f$ in the proof of Theorem 2 has 6 children. From the above theorem, the approximation ratio of the obtained algorithm BI($1-\frac{5}{5e+1}$) becomes

$$\frac{\sqrt{(5e+1)^2+25}-5}{5e+1} \geq 0.7144139.$$

4. Conclusion

In this paper, we proposed two techniques which improve existing approximation algorithms. If we apply the implicit enumeration technique to an α-approximation algorithm, the approximation ratio increases to $(1-\frac{(k-1)(1-\alpha)}{(k-1)+(1-\alpha)})$ where k denotes the number of children of the root. Here we note that $\lim_{k\to\infty}(1-\frac{(k-1)(1-\alpha)}{(k-1)+(1-\alpha)}) = \alpha$. If we employ a polynomial time α-approximation algorithm, Algorithm IE(α) is also a polynomial time algorithm. By applying the implicit enumeration technique recursively, the approximation ratio tends to 1 and the computational time grows exponentially.

The backward induction technique improves an approximation algorithm, if a given tree is ternary. By applying to a polynomial time approximation algorithm, the backward induction technique also gives a polynomial time approximation algorithm. When a given tree is k-ary and $k \geq 4$, the backward induction technique does not improve the approximation ratio. Even if a given tree is 4-ary, recursive application of the backward induction technique does not improve the approximation ratio, since subtrees dealt in the backward induction technique are not 4-ary any longer.

References

[1] E. Anshelevich, D. Chakrabarty, A. Hate, and C. Swamy, "Approximation algorithms for the firefighter problem: Cuts over time and submodularity," Proc. 20th International Symposium on Algorithms and Computation (ISAAC'09), vol.5878, Lect. Notes Comput. Sci., pp.974–983, Springer, 2009.
[2] L. Cai, E. Verbin, and L. Yang, "Firefighting on trees: $(1-1/e)$-approximation, fixed parameter tractability and a subexponential algorithm," Proc. 19th International Symposium on Algorithms and Computation (ISAAC'08), vol.5369, Lect. Notes Comput. Sci., pp.258–269, Springer, 2008.
[3] S. Finbow, A. King, G. MacGillivray, and R. Rizzi, "The firefighter problem for graphs of maximum degree three," Discrete Mathematics, vol.307, no.16, pp.2094–2105, 2007.
[4] B. Hartnell, "Firefighter! An application of domination," 24th Manitoba Conference on Combinatorial Mathematics and Computing, University of Minitoba, Winnipeg, Cadada, 1995.
[5] B. Hartnell and Q. Li, "Firefighting on trees: How bad is the greedy algorithm?," Congressus Numerantium, vol.145, pp.187–192, 2000.

Yutaka Iwaikawa is presently a master course student at Chuo University. His research interests are combinatorial optimization and graph theory.

Naoyuki Kamiyama received B.Eng., M.Eng., and Dr.Eng. degrees from Kyoto University in 2004, 2006, and 2009, respectively. He is now an assistant professor in Chuo University. His research interests include the design and analysis of combinatorial algorithms. He is a member of the Information Processing Society of Japan, Architectural Institute of Japan, and the Operations Research Society of Japan.

Tomomi Matsui received B.E., M.E. and Dr. of Science from Tokyo Institute of Technology. He is now a professor at Chuo University. His areas of interests include combinatorial optimization and enumeration algorithm.