Linear Time Algorithms for k-cutwidth Problem

Maw-liwa Chen and Sing-Ling Lee

Institute of Computer Science and Information Engineering
National Chung Cheng University
Chaiyi, Taiwan 62107, ROC

Abstract

The Min Cut Linear Arrangement problem is to find a linear arrangement for a given graph such that the cutwidth is minimized. This problem has important applications in VLSI layout systems. It is known that this problem is NP-complete when the input is a general graph with maximum vertex degree at most 3. In this paper, we will first present a linear time algorithm to recognize the small cutwidth trees. The approach we used in this algorithm can then be easily extended to recognize the general graphs with cutwidth 3 in \(O(n)\) time.

1. Introduction

A linear arrangement (also called numbering, labeling, or layout) of an undirected graph \(G = (V, E)\), \(|V| = n\), is a 1-1 function \(f : V \rightarrow \{1, 2, \ldots, n\}\). The cutwidth of \(G\) under a linear arrangement \(f\), denoted by \(cw(G, f)\), is

\[
\max_{1 \leq i < n} \left| \{(u, v) \in E | f(u) < i < f(v)\} \right|.
\]

The cutwidth of a graph \(G\), denoted by \(cw(G)\), is the minimum of \(cw(G, f)\) taken over all possible linear arrangement \(f\). The \(k\)-cutwidth problem is to determine whether or not there exists a linear arrangement of the vertices such that the graph has cutwidth at most \(k\). Clearly, the \(k\)-cutwidth problem is the decision version of the Min Cut Linear Arrangement Problem. Example of the Min Cut Linear Arrangement problem is shown in Fig. 1.1.

![Fig. 1.1](image)

(a) Source graph. (b) \(f(h) = 1, f(a) = 2, f(b) = 3, f(d) = 4, f(c) = 5, f(e) = 6, f(i) = 7, f(g) = 8\), and \(cw(G, f) = 3\).

---

* This work was supported by the National Science Council, Taiwan, R.O.C. Grant NSC 80-0404-E194-03
The Min Cut Linear Arrangement problem has many applications in VLSI CAD design. We can use the graph to abstract the representation of the circuit. The vertices of the graph are the circuit elements and the edges form the needed interconnection between them. In some approaches to VLSI design, the circuit elements must be first placed in rows or in a single line such that the cutwidth is minimized [6,12,14,15]. It is known that the MIN CUT problem is NP-complete even when restricted to graphs of degree at most 3 [12]. However, deciding whether a given graph with \( n \) vertices has cutwidth at most \( k \) or not can be done in time \( O(n^k) \) [8]. If the graph is a tree \( T \), then Yannakakis has found an \( O(n \log n) \) to construct \( ew(T) \) [17]. In this paper, we will first present a linear time algorithm to recognize the small cutwidth trees. The approach we used in this algorithm can then be easily extended to recognize the general graphs with cutwidth 3 in \( O(n) \) time.

In the next section, we introduce some properties of the Min Cut Linear Arrangement problem on trees. Section 3 outlines the main ideas of our approach and the description of the algorithm to solve the 3-cutwidth problem on trees. The computational complexity of the algorithm is also analyzed in this section. In section 4, the extension of our method given in section 3 will be discussed. In section 5, we use similar approach to solve the 3-cutwidth problem on general graphs. Finally, we give the conclusion in section 6.

2. Preliminaries

In this section we will introduce several properties of the Min Cut Linear Arrangement problem on trees given in [4]. The approaches of our algorithm are based on these properties.

Let \( T = (V, E) \) be a tree, \( |V| = n \). Define \( P_f \) as the path connecting the two vertices \( u,v \) with \( f(u) = 1 \) and \( f(v) = n \) under the Min Cut Linear Arrangement \( f \) in \( T \) and \( T - P_f \) as the forest formed by removing the edges of \( P_f \) from \( T \) (but let the vertices of \( P_f \) stay). Then there always exists an optimal linear arrangement \( f^* \) with \( cw(T, f^*) = ew(T) \) satisfying the following properties:

1. The leaf property: The two vertices \( u,v \) with \( f^*(u) = 1 \) and \( f^*(v) = n \) are leaves. (See Fig. 2.1(b).)
2. The monotone property: Suppose \( P_f \) has vertices \( v_0, v_1, \ldots, v_t \) with \( v_i \) adjacent to \( v_{i+1} \). Then \( f^*(v_i) < f^*(v_{i+1}) \) for \( i = 0, 1, 2, \ldots, t-1 \) or \( f^*(v_t) > f^*(v_{i+1}) \) for \( i = 0, 1, 2, \ldots, t-1 \). (See Fig. 2.1(b).)
3. The block property: Suppose \( T - P_f \) has trees \( T_1, T_2, \ldots, T_m \) \((m \geq 1)\). Then for each vertex \( u \in T_i \) \((1 \leq i \leq m)\), there exist no two vertices \( v, w \in T_j \) \((j \neq i)\) such that \( f^*(v) < f^*(u) < f^*(w) \), i.e. all vertices in \( T_i \) \((1 \leq i \leq m)\) are labeled by a set of consecutive integers. (See Fig. 2.1(b) and (d).)
4. The hereditary property: The induced labeling for each subtree \( T_i = (V_i, E_i) \) of \( T - P_f \) is optimal with respect to the Min Cut Linear Arrangement. (The induced labeling \( f_i \) of \( f^* \) on \( T_i \) is the one-to-one mapping from vertex set \( V_i \) to the set \( \{1, 2, \ldots, |V_i|\} \) such that for any edge \( \{u, v\} \in E_i \), \( f_i(u) < f_i(v) \) if \( f^*(u) < f^*(v) \).) (See Fig. 2.1(b) and (d).)

Directly from the above properties, we now have the following lemma.

**Lemma 2.1.** Suppose \( f^* \) is optimal with respect to the Min Cut Linear Arrangement for \( T \). Then \( cw(T) = 1 + \max_{1 \leq i \leq m} ew(T_i) \) where \( T_i \) is a subtree in \( T - P_f \).

At the end of this section, we will give an example that will show these properties and lemmas in this section. The detail is shown in Fig. 2.1.