1986

Two-Layer Channel Routing with Vertical Unit-Length Overlap

Shaodi Gao

Susanne E. Hambrusch
Purdue University, seh@cs.purdue.edu

Report Number:
86-578

TWO-LAYER CHANNEL ROUTING WITH
VERTICAL UNIT-LENGTH OVERLAP

Shaodi Gao
Susanne Hambrusch

CSD-TR #578
February 1986

Two-Layer Channel Routing with Vertical Unit-Length Overlap

Shaodi Gao
Fachbereich 10
Universitat des Saarlandes
D-6600 Saarbrucken, West Germany.

*Susanne Hambrusch**
Department of Computer Sciences
Purdue University
West Lafayette, IN 47907, USA.

Abstract

We show that any n-net 2-terminal channel routing problem of density d can be wired on a two-layer grid of width $w = d + O(d^{2/3})$ when vertical wire segments are allowed to overlap for a distance of length 1. This is a considerable asymptotic improvement over the best known, and optimal, channel width of $2d-1$ for models in which no vertical overlap is allowed [RBM, PL]. Our result also improves the $3d/2 + O(1)$ channel width achieved by a recent algorithm [G] for the same vertical overlap model. The algorithm presented in this paper produces at most 4 overlaps of unit length between any two nets, uses $O(n)$ contacts, and can be implemented to run in $O(nd^{2/3})$ time. We also generalize the algorithm to multi-terminal channel routing problems for which our algorithm uses a width of $w = 2d + O(d^{2/3})$.

Key Words

Analysis of Algorithms, channel routing, channel width, density, overlap.

* This work was supported by the Office of Naval Research under Contract N00014-84-K-0502 and by the National Science Foundation under Grant DMC-84-13496.

1. Introduction

The Channel Routing Problem (CRP) is the problem of connecting terminals belonging to signal nets and located on two opposite sides of a rectangular channel so that the wiring uses minimum area. Ideally, other cost measures (e.g., the number of contacts) should also be minimized in the routing process. More formally, in a CRP we are given a channel of length l and n nets, $N_1, \cdots, N_n$, where net $N_i = (T_i, B_i)$ and T_i (resp. B_i) contains the positions of the terminals of net N_i on the top (resp. bottom) row of the channel, $T_i, B_i \subseteq \{1, \cdots, l\}$, $T_i \cap T_j = \emptyset$ and $B_i \cap B_j = \emptyset$ for $i \neq j$. If $|T_i| = |B_i| = 1$, $1 \leq i \leq n$, we call the problem a 2-terminal CRP, otherwise a multi-terminal CRP.

Because of the importance of channel routing in the design of layout systems [HS, R], numerous heuristics and approximation algorithms have been proposed for a number of wiring models [BBL, D, H, G, PL, RBM, RF, SP, YK]. In the 2-layer knock-knee model (in which two wires on different layers are allowed to share a corner [KM, MP, PL, RBM]) any 2-terminal (resp. multi-terminal) CRP can be solved using a channel width of $2d-1$ (resp. $4d-1$), where d is the density of the CRP which will be defined later. Leighton has shown that there exists a class of 2-terminal CRPs that require a width of $2d-1$ [L].

The wiring model used in this paper differs slightly from the knock-knee model. Instead of allowing knock-knees, we allow two wires on different layers to run on top of each other for one vertical unit. The two layers are used in a *quasi-directional* fashion (which is stricter than in the knock-knee model): All the horizontal wire segments lie in layer 1, and all the vertical wire segments, except the ones that could participate in a vertical overlap, lie in layer 2. We describe an algorithm that solves any n-net 2-terminal CRP using a channel width of $d + O(d^{2/3})$, $O(n)$ vertical unit overlaps, and $O(n)$ contacts. Our results is actually more general: We show that for any integer k, a channel width of $\lceil \frac{k+1}{k} (d+4k^2+2) \rceil + 1$ can be achieved, which is minimized for $k = d^{1/3}/2$. Our algorithm generalizes the one presented in [G]. For the case of multi-terminal CRPs we show how to achieve a width of $2d + O(d^{2/3})$.

The channel width required by our algorithms does only depend on the density, which is the maximum over the number of all nets that have to cross from column i to column $i+1$. It is defined as

$$d = \max_i \{ | \{ N \mid N \text{ is a net with } min(T \cup B) \leq i \text{ and } max(T \cup B) \geq i+1 \} | \}$$

It is easy to see that even when vertical unit overlap is allowed, the density is a lower bound on the channel width. In the directional wiring model [BR, BBL], which is also called the Manhattan Model, the width does not only depend on the density and could be $n^{1/2}$ for a CRP of density 2 [BR].

We next describe our 2-layer grid wiring model in more detail. The vertical lines of the grid are called the *columns*, and the horizontal lines are called the *tracks*. The tracks (resp. columns) are numbered $0, \cdots, w+1$ (resp. $1, \cdots, l$) from top to bottom (resp. from left to right), and w is the *width* of the channel. Every intersection of a track and a column forms a *grid point*. A solution to a 2-terminal CRP contains for every net N_i of a wire w_i starting at the terminal on track 0, consisting of horizontal and vertical wire segments (whose end points are grid points), and ending at the terminal on track $w+1$. (A solution to a multi-terminal CRP is defined analogously.) Horizontal wire segments are assigned to layer 1, and vertical segments (except one of the two of a vertical overlap) are assigned to layer 2. In order to change from layer 1 to layer 2 at a grid point (x,y), which is on track x in column y, the wire makes a *contact* at (x,y). No two wires are allowed to run through the same grid point in the same layer (such a situation is called a short circuit). Two wire segments on different layers can cross each other at a grid point (in a right angle) and can share a vertical segment of length 1; i.e., wires w_i and w_j can both contain the vertical segment $((x,y),(x+1,y))$ provided one uses it on layer 1 and the other on layer 2, and the vertical segments $((x+1,y),(x+2,y))$ and $((x-1,y),(x,y))$ contain no overlap.

Section 2 describes the routing algorithm for 2-terminal CRPs, and in Section 3 we present its analysis. In Section 4 we discuss the modifications to be done for multi-terminal CRPs.

2. The Algorithm for the 2-Terminal Problem

In this section we show how to solve any 2-terminal CRP of density d on a channel of width $w = gk+g+1$, where $k \geq 1$ and $g = \lceil \frac{d+2}{k} + 4k \rceil$. The algorithm divides the channel into g groups, each group consisting of k consecutive tracks, also called *layout tracks*. The groups (resp. tracks in a group) are numbered $1, \cdots, g$ (resp. $1, \cdots, k$) with 1 being the top-most group (resp. top-most track in a group). We refer to track x in group i as track (i,x), $1 \leq i \leq g$, $1 \leq x \leq k$.

Between tracks (i,k) and $(i+1,1)$, $1 \le i < g$, above track (1,1), and below track (g,k) there is one *additional track*, track a_i, $0 \le i \le g$. Let (i,x) and (j,y) be two layout tracks. Then $(i,x)<(j,y)$ if either $i<j$ or, if $i=j$, then $x<y$.

The algorithm determines the wires of the nets by scanning the channel once, from left to right. Before describing the algorithm we define some additional terminology. Let $N_i = (t_i, b_i)$, $1 \le i \le n$, be the 2-terminal nets of the CRP. A net N_i is called *extended* in column c if both terminals are to the left of column c (i.e., $\max(t_i,b_i)<c$) and net N_i still occupies tracks between columns $c-1$ and c. An extended net N_i is *closed* in column c if the algorithm makes a vertical wire segment in column c that connects the tracks occupied by net N_i (an extended net occupies exactly two tracks). A net N_i is called *active* if it has exactly one terminal to the left of column c (i.e., $\min(t_i,b_i)<c$ and $\max(t_i,b_i) \ge c$). If $\max(t_i,b_i)<c$ and net N_i is not extended, then we say that net N_i has been *completed.*

Assume that the input consists of two sorted lists: one containing the n nets sorted according to their top terminals t_i, and one containing the nets sorted according to their bottom terminals b_i. The algorithm scans, simultaneously, these two input lists and the channel, in a left-to-right, column-by-column manner. Assume the scan is currently at column c. Then the wires to be placed in columns $1, \cdots, c-1$ and on the tracks up to column c have already been determined. The horizontal wires going from column $c-1$ to column c belong either to active or extended nets. Extended nets are divided into two categories: closable and non-closable ones. The precise definition of a closable and a non-closable net will be given after the informal outline of the algorithm. We will also show in the next section that if there are $2k^2+1$ extended nets present in any column, then at least one of them is closable.

We now give the overall outline of the algorithm. We assume that every column c contains an upper and a lower terminal (if this is not the case, the modifications to be done are straightforward).

Outline of the Algorithm

(1) **for** column $c = 1, 2, \ldots$ **do**

(2) **if** there exists a closable extended net, say net α

(3) **then** close net α, and handle the wiring of the nets containing the upper (resp. lower) terminal; this will be described in detail below

(* in this case no previously empty layout tracks are used; the wiring done in this step changes the status of some extended nets (i.e., from closable to non-closable and vice-versa) *)

(4) **else** Choose two available layout tracks (s,p) and (t,q), $(s,p)<(t,q)$, and place the net containing the upper terminal on track (s,p) and the net containing the lower terminal on track (t,q)
(* there are at most $2k^2$ extended nets present and thus at most $d+4k^2$ layout tracks are occupied; since the algorithm has a total of $d+4k^2+2$ layout tracks, at least two of them are available *)
fi
od.

The definition and the purpose of a closable net will become apparent after describing line (3) in more detail, which is done next. Assume net α occupies the layout tracks (i,x) and (j,y), $(i,x)<(j,y)$. We close net α by making a vertical segment in layer 2 which requires two contacts in column c. See Figure 2.1.

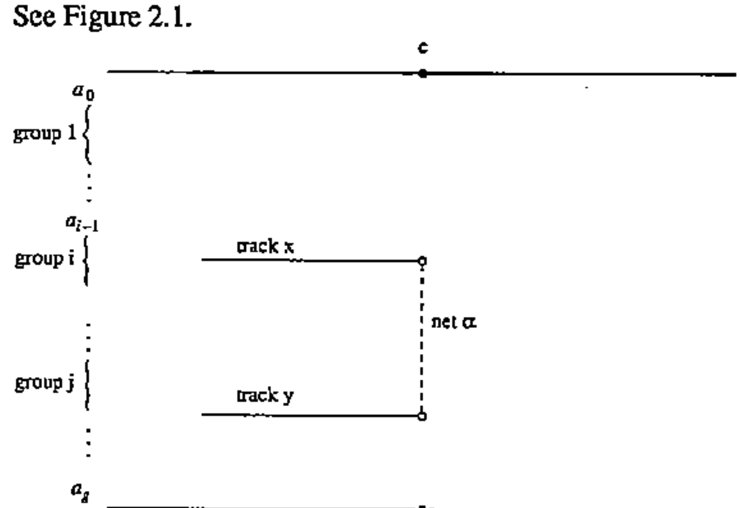
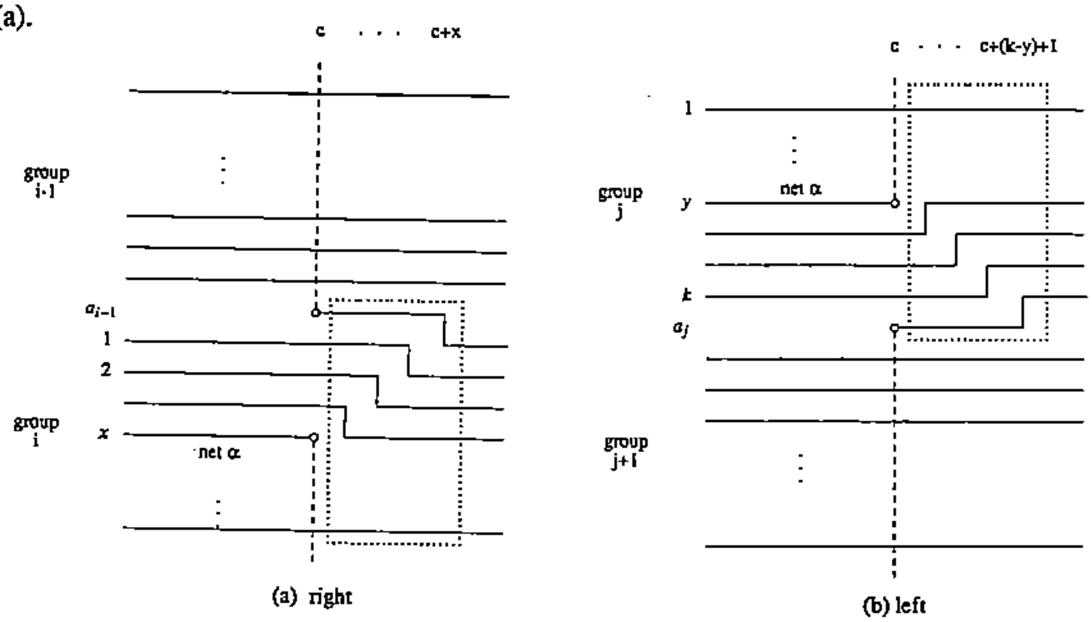


Figure 2.1

Let the upper terminal in column c belong to net β. If net β is an active net and on a track above track (i,x), the algorithm completes net β. Otherwise (i.e., net β is either active on a track below track (i,x) or not yet active), the algorithm shifts the nets currently on tracks $(i,1), \cdots, (i,x-1)$ onto tracks $(i,2), \cdots, (i,x)$ by letting the net originally on track (i,l) make its vertical unit segment in column $c+x-l$, $1 \le l \le x-1$. The wire from the upper terminal in column c (which belongs to net β) is brought onto track $(i,1)$ by using the additional track a_{i-1} up to column $c+x$. We say that the algorithm creates a *right knock-knee region* in column c on track (i,x), which consists of all the tracks in group i and the additional track a_{i-1} in columns c, $c+1$, ... , $c+x$. See Figure

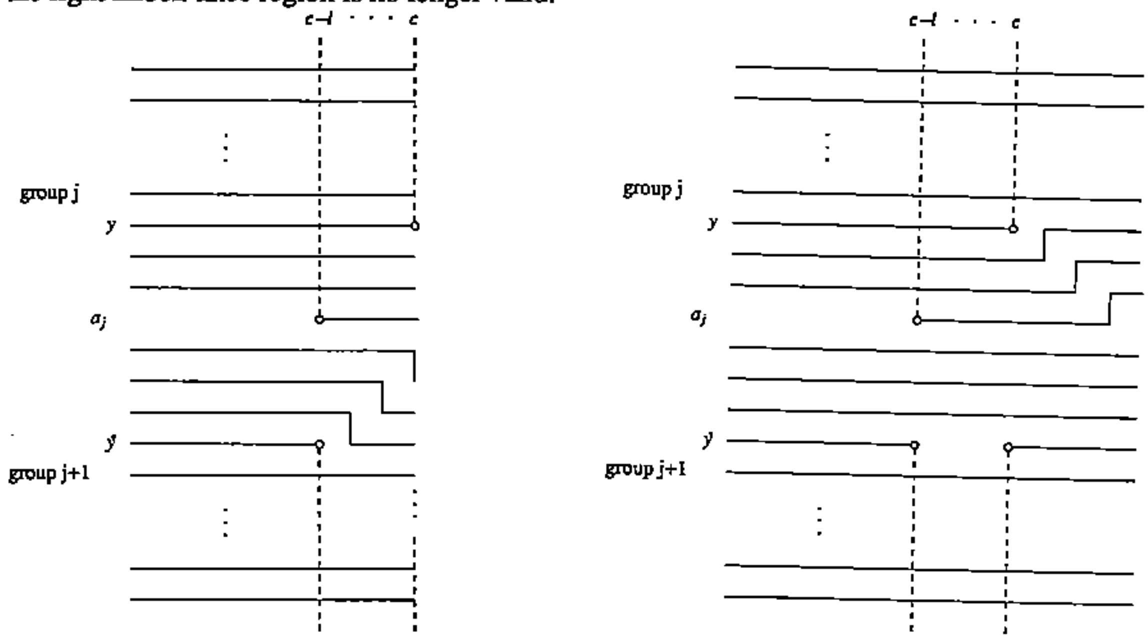
2.2(a).



knock-knee region (indicated by .....)
Figure 2.2

Analogously, if column c contains a lower terminal we may need to create a *left knock-knee region.* In such a case the nets on tracks $(j,k), \cdots, (j,y+1)$ are shifted onto tracks $(j,k-1), \cdots, (j,y)$ and the additional track used in the knock-knee region is a_{j+1}. See Figure 2.2(b). The rules shown in Figure 2.2 are the basic wiring rules. If, for example, a right knock-knee region is created in column c on track (i,x), then no other knock-knee region that overlaps this one can be created. This motivates the definition of a closable net. An extended net α occupying tracks (i,x) and (j,y) is called *closable* if the two knock-knee regions created in line (3) (i.e., in the process of closing net α and placing the wires of other nets onto the layout tracks used by net α), do *not* overlap with any previously created knock-knee regions in groups i and j. If such an overlap exists, net α is called *non-closable.*

The definition of a closable net has one exception: A region created in a group i can overlap with a region in an adjacent group; i.e., two such regions can share the same additional track. The wiring in this situation is done as follows. Assume we want to form a left knock-knee region on track (j,y) in column c, and a right knock-knee region has been created on track $(j+1,\hat{y})$ in column $c-l$, $l \leq \hat{y}$. See Figure 2.3(a). Both knock-knee regions share track a_j. The lower wire of the right knock-knee region created in column $c-l$ and the upper wire of the left knock-knee region both belong to closed nets. The algorithm rewires as shown in Figure 2.3(b): It puts the lower wire creating the left knock-knee region on track $(j+1,\hat{y})$. The upper wire of the right knock-knee region runs first on the additional track a_j and, from column c on, is handled like the lower wire of the left knock-knee region. In column c we thus create a left knock-knee region and the right knock-knee region is no longer valid.



(a) right knock-knee region created at $(\hat{y},c-l)$ Figure 2.3 (b) rewiring

This completes the description of the algorithm. Before analyzing its performance, we briefly describe an improvement to the wiring rules. (Although this does not improve the asymptotic performance, it may be useful for practical purposes.) It is easy to see that the definition of a closable net is rather strict, and we now modify the wiring rules so that two knock-knee regions of different type (i.e., left and right) are allowed to overlap. In order to do so, the algorithm has to, as done in the rule shown in Figure 2.3, 'look back' at previous columns and to possibly change the wiring already done. But doing so does not increase the time complexity. We only discuss the modification to be done when a right knock-knee region was created in column $c-l$ on track (i,x); the case for left knock-knee regions is handled analogously. Assume we want to create a left knock-knee region in column c on track (i,z), which overlaps with the right knock-knee region.

(A) If $z>x$, then we can create a left knock-knee region according to the rule shown in Figure 2.2(b).

(B) If $z \leq x$, we distinguish 3 cases:

(i) $x-z>l$. In this case the upper wire creating the left-knock knee region did already move onto track $z+1$. The algorithm can form the left knock-knee region according to the basic rule.

(ii) $x-z=l$. In this second case we place the lower wire creating the left knock-knee region on the track $z+1$ (which is the track the upper wire of the left knock-knee region would have used). The algorithm creates no left knock-knee region and the right knock-knee region remains as before. See Figure 2.4(a).

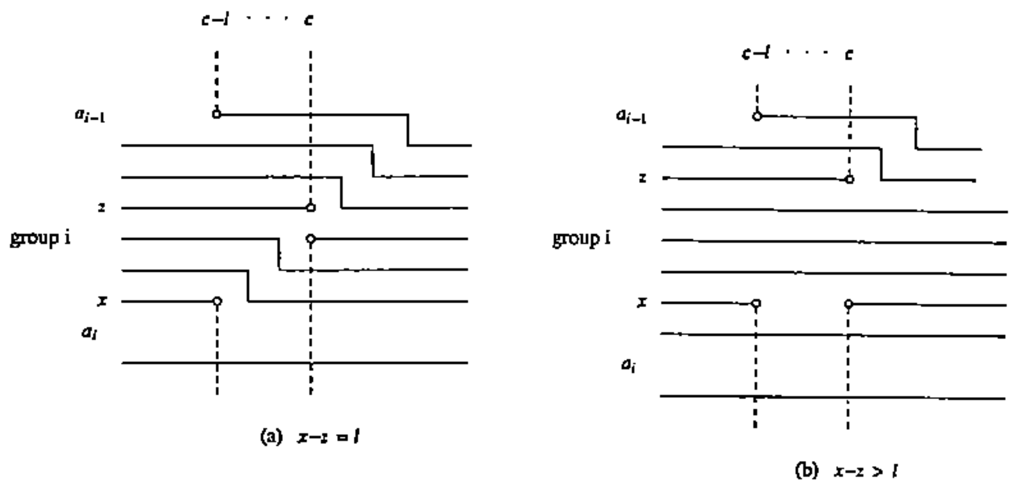


Figure 2.4

(iii) $x-z>l$. In this case we wire as shown in Figure 2.4(b): The wires on tracks $x+1, \cdots, z-1$ do not change track (i.e., we 'undo' the wiring). The lower wire of the left knock-knee region is put on track (i, x). No left knock-knee region is created and the right knock-knee region may now be smaller.

This concludes the discussion of the modifications and in the next section we analyze the performance of the algorithm.

3. Analysis of the Algorithm

The algorithm described in the previous section uses $\lceil \frac{k+1}{k} (d+4k^2+2) \rceil + 1$ tracks and we now show that it successfully wires any n-net 2-terminal CRP of density d. We also discuss the amount of overlap used and the number of contacts made during the algorithm, and how to implement the algorithm to run in $O(nd^{2/3})$ time.

In order to prove correctness of the algorithm we need to show that, when processing column c, the algorithm either finds a closable net or that there are two available layout tracks. Recall that an active net occupies one and an extended net occupies two tracks. Thus, between columns $c-1$ and c, there are at most d tracks occupied by active nets, and the remaining $4k^2+2$ layout tracks are either available or occupied by extended nets. In every column the algorithm constructs at most two knock-knee regions. The knock-knee regions created in columns $c-k, \cdots, c-1$ were formed in at most $2k$ distinct groups (recall that knock-knee regions in adjacent groups can overlap), and these $2k$ groups can contain at most $2k^2$ extended nets (all of which are still present in column c). If, in column c, the $4k^2+2$ layout tracks contain more that $2k^2$ extended nets, then line (2) of the algorithm finds a closable extended net. If there are fewer than (or exactly) $2k^2$ extended nets each of which is non-closable, then there are at least 2 available layout tracks in the channel and line (4) of the algorithm can be executed. Thus, column c of the algorithm can always be wired and correctness follows.

It is obvious from the wiring rules given in Figures 2.2, 2.3, and 2.4 that the algorithm produces overlap only in the vertical direction. Once a net occupies a track in a group (or, in the case of an extended net, two tracks in possibly two groups) the net does not change groups, although it may change tracks within a group. Whenever a wire changes track it may produce vertical overlap of length one with a wire of another net. Since every net contains at most 3 vertical segments in layer 2 (which is the layer exclusively used for vertical connections), two nets can overlap with each other for at most 6 vertical units. One can actually show that two nets can only overlap with each other at most 4 times. This comes from the fact that wires do not change their group and that they remain in the same relative order to each other within this group. Hence, the total number of vertical unit overlap segments produced by the algorithm is $O(n)$. Note that the wire segment of a net on layer 1 can possibly overlap with a wire segment of a different net every k columns. It is furthermore clear from the wiring rules that our algorithm uses $O(n)$ contacts: Every net uses either 0, 2, or 4 contacts.

While the algorithm works for arbitrary values of k, the number of tracks is minimized for $k = d^{1/3}/2$. In this case the channel width $w = d + 4d^{2/3} + O(d^{1/3})$ which, in the asymptotic sense is better than the channel width required by other known algorithms. For the case of $k = d^{1/3}/2$, we show how to implement the algorithm so that it runs in $O(nd^{2/3})$ time. Observe that this time is optimal in the sense that there are CRPs whose explicit description of the wiring produced by the algorithm (i.e., in the wiring every change in a track of a net is explicitly stated) is of length $O(nd^{2/3})$.

During the scan the algorithm keeps the extended nets in a list. For every extended net α we record $(\alpha,(i,x),(j,y))$ with $(i,x)<(j,y)$ in this list. In order to achieve $O(nd^{2/3})$ time, the list does not need to be kept sorted and thus a newly created extended net is added in constant time. The algorithm also keeps two arrays LR and RR, both of size $g = \lceil \frac{2(d+2)}{d^{1/3}} + 2d^{1/3} \rceil$, where $LR(i)$ (resp. $RR(i)$) contains the column and track number of the last left (resp. right) knock-knee region created in group i.

When processing column c the algorithm scans through the list of extended nets until it either finds a closable net or the list is exhausted. Since there are at most $2k^2$ closable nets, line (2) of the algorithm takes $O(d^{2/3})$ time. The algorithm has to record the shifts in the tracks for the nets involved in the knock-knee region, which can be done in $O(d^{1/3})$ time. Note that 'undoing' the wiring and updating the information about knock-knee regions when necessary can also be done in $O(d^{1/3})$ time. Since we only need to consider columns that contain terminals and/or close extended nets, the total running time of the algorithm is $O(nd^{2/3})$.

We are now ready for the main result about 2-terminal CRPs:

Theorem 1. Any n-net 2-terminal CRP of density d can be wired on a channel of width $w = d + O(d^{2/3})$. The wiring contains at most 4 vertical overlaps of length 1 between any two nets, uses $O(n)$ contact points, and can be determined in $O(nd^{2/3})$ time.

4. Multi-Terminal Nets

The basic idea of the algorithm described in Section 2 can be extended to handle multi-terminal CRPs. In order to do so we divide the channel into two vertical strips, the upper and the lower strip. Each strip consists of $\lceil \frac{k+1}{k}(d+4k^2+2) \rceil + 1$ tracks and is organized as the channel for the 2-terminal CRP. A wire segment originating at a terminal on the top row of the channel is always placed on a track in the upper strip, and a wire segment originating at a terminal on the bottom row is always placed on a track in the lower strip.

The algorithm also scans the channel from right to left, column by column. During the algorithm a net can occupy two or more tracks in a strip (such a net is called a *collapsible net*). When processing column c the algorithm first determines if there exists an extended net which uses tracks in both the upper and the lower strip. If such a net exists and knock-knee regions can be created without violating the wiring rules (which are as in Section 2), the algorithm closes this net. If no such net exists, the algorithm considers the upper and the lower strip separately. In each strip it either finds a collapsible net for which it can make a vertical wire segment that frees up some of the tracks occupied by this net, or it finds an available track which is then used for the wire segment originating at the terminal on the same side. The algorithm thus uses $2\,(\lceil \frac{k+1}{k}(d+4k^2+2)\rceil+1)$ tracks for a multi-terminal CRP of density d. The proof of correctness of the algorithm is similar to the one for 2-terminal CRPs and is omitted. Again, $k = d^{1/3}/2$ minimizes the channel width, and we can thus state the following theorem.

Theorem 2. Any n-net multi-terminal CRP of density d can be wired on a channel of width $2d + O(d^{2/3})$. The algorithm makes $O(m+l)$ contacts and $O(m+l)$ unit length overlaps, where m (resp. l) is the total number of terminals positioned on the top (resp. bottom) row of the channel.

5. Conclusions

We presented a new two-layer channel routing algorithm for the quasi-directional model with vertical unit overlap. We showed that within this model any 2-terminal CRP can be solved in a channel of width $d + O(d^{2/3})$. This algorithm does not only achieve a smaller channel width than previously known algorithms †, it also uses fewer contacts ($O(n)$ compared to $\Omega(dn)$ for the knock-knee model [RBM]). We described how to generalize the algorithm for the case of multi-terminal CRPs for which it achieves a channel width of $2d + O(d^{2/3})$.

† Berger et al. [BBBL] have recently developed an algorithm that achieves a channel density of $d + O(d^{1/2})$ for the same model.

References

[BBBL]B. Berger, M. Brady, D. Brown, F.T. Leighton, personal communication.

[BBL] B.S. Baker, S.N. Bhatt, F.T. Leighton, 'An Approximation Algorithm for Manhatten Routing', *Proceedings of the 15th Annual ACM Symp. on Theory of Computing*, pp 477-486, 1983.

[BR] D.J. Brown, R.L. Rivest, 'New Lower Bounds for Channel Width', *Proceedings of the CMU Conf. on VLSI Systems and Comp.*, pp 153-159, 1981.

[D] D.N. Deutsch, 'A Dogleg Channel Router', *Proceedings of the 13th IEEE Design Automation Conf.*, pp 425-433, 1976.

[G] S. Gao, 'An Algorithm for Two-layer Channel Routing', *Proceedings of 2-nd Ann. Symp. on Theoretical Aspects in Comp. Sc.*, pp 151-160, 1985.

[H] S.E. Hambrusch, 'Channel Routing Algorithms for Overlap Models', *IEEE Trans. on CAD*, Vol. cad-4, pp 23-31, Jan. 1985.

[HS] A. Hashimoto, J. Stevens, 'Wire Routing by Optimizing Channel Assignment within Large Apertures', *Proc. of 8-th Design Aut. Conf.*, pp 155-169, 1971.

[KM] M. Kaufmann, K. Mehlhorn, 'Routing Through a Generalized Switchbox', *Lecture Notes in Computer Science*, Nr. 194 (ICALP 85), Springer Verlag, pp 328-333, 1985.

[L] F.T. Leighton, 'New Lower Bounds for Channel Routing', unpublished manuscript, 1981.

[MP] K. Mehlhorn, F.P. Prepatara, 'Routing Through a Rectangle', *JACM*, Vol. 33, Nr. 1, pp 60-85, 1986.

[PL] F.P. Preparata, W. Lipski, 'Three Layers are enough', *Proceedings of the 23rd Annual IEEE Foundations of Comp. Sc. Conf.*, pp 350-357, 1982.

[R] R.L. Rivest, 'The PI - Placement and Interconnect - System', *Proc. of 19-th Design Automation Conf.*, pp 475-481, 1982.

[RBM]R.L. Rivest, A.E. Baratz, G. Miller, 'Provably Good Channel Routing Algorithms', *Proceedings of the CMU Conf. on VLSI Systems and Computations*, pp 153-159, 1981.

[RF] R.L. Rivest, C.M. Fiduccia, 'A "Greedy" Channel Router', *Proceedings of the 19t-th IEEE Design Automation Conf.*, pp 418-424, 1982.

[SP] M. Sarrafzadeh, F.P. Prepatata, 'Compact Channel Routing of Multi-terminal Nets', to appear in *Annals of Discrete Mathematics*.

[YK] T. Yoshimura, E.S. Kuh, 'Efficient Algorithms for Channel Routing', *IEEE Trans. on CAD*, pp 25-35, 1982.