PERIODICA POLYTECHNICA SER. EL. ENG. VOL. 46, NO. 3–4, PP. 175–194 (2002)

ROUTING WITH MINIMUM WIRE LENGTH IN THE MANHATTAN MODEL IS NP-COMPLETE[1]

Tibor Szkaliczki

Computer and Automation Research Institute
H-1111 Budapest, Lágymányosi u. 11., Hungary
phone: +361 279 6000/ 7347
e-mail: sztibor@sztaki.hu
http://pele.ilab.sztaki.hu/~sztibor

Received: July 15, 2003

Abstract

The present paper concentrates on one of the most common routing models, on the Manhattan model where horizontal and vertical wire segments are positioned on different sides of the board. While the minimum width can be found in linear time in the single row routing, apparently there was no efficient algorithm to find the minimum wire length. We showed before that this problem is NP-complete in the dogleg-free case but the complexity of the problem was still unknown in the general case. In this paper we modify the construction applied in the former proof in order to show the NP-completeness of routing with minimum wire length in the Manhattan model without any restrictions.

Keywords: single row routing, VLSI, NP-complete problems, minimum length.

1. Introduction

VLSI routing plays a relevant role in the design of integrated circuits. The input of the VLSI circuit layout process is a description of a circuit. The purpose is to determine the exact placement of the circuit elements that satisfies the technological requirements and that minimizes certain cost criteria. The layout process consists of several phases such as placement, global routing, local routing and compaction. During global routing, the total routing region is partitioned into subregions (usually rectangles) and it is decided through which routing subregions individual wires will run. During detailed or local routing, the exact course of the wires and the sizes of the routing subregions are determined. The detailed-routing problem is solved subregion by subregion. Routing with minimum total wire length in the Manhattan model is one of the most common problems in detailed routing. The efficiency of the algorithms is crucial, because the number of the terminals to be interconnected is extremely large.

Let us give some basic definitions of detailed routing. The points to be interconnected are called *terminals*. Routing within a rectangle is a basic problem

[1]This work was supported by the Hungarian National Foundation for Scientific Research (Grant No. OTKA T 42559)

of the VLSI design. In case of *single row routing* all terminals appear only on one side of the rectangle. This is a special case of the *channel routing* where all terminals are located either at the upper or the lower boundary of the routing region. A *net* is a collection of terminals. An instance of the problem is a set of pairwise disjoint nets. It is *dense* if every boundary point belongs to some nets. The solution of a routing problem is a set of subgraphs (wires) where each subgraph connects all the terminals of the corresponding net under the conditions of the wiring model. In the *Manhattan model* wires run on a rectangular grid and horizontal and vertical wire segments are positioned on different sides (*layers*) of the board. The *tracks* and the *columns* are the horizontal and the vertical lines of the grid, respectively. The number of tracks is called the *width* of the channel. In a restricted version of the Manhattan model each wire could occupy only one track. This model is called the *dogleg-free* model. If *doglegs* are permitted then wires can switch from one track to another. We are interested in the complexity of finding the minimum wire length solution of the given routing problem in the Manhattan model. LENGAUER (1990) presents a detailed exposition of the routing in the Manhattan model. *Fig. 1* shows a routing problem and its solution.

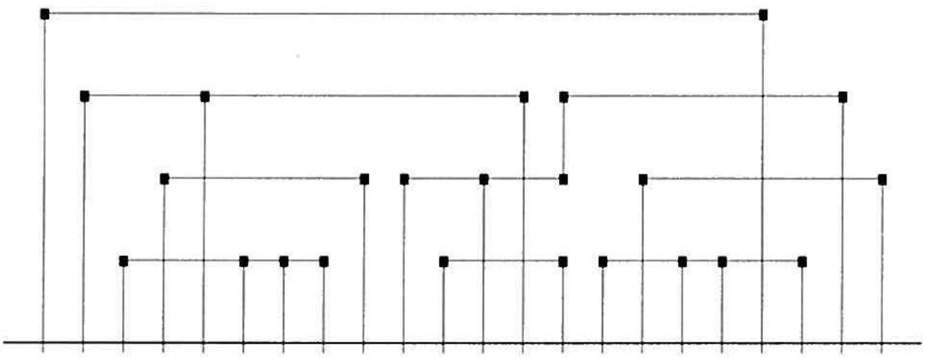

Fig. 1. Single row routing

The minimum width can be found in linear time in the single row routing in the Manhattan model. ([1], see also [5]). LA PAUGH (1980) proved in the dogleg-free case and SZYMANSKI (1985) in the general case that the channel routing is $\mathcal{NP}$-complete in the Manhattan model. We proved in our earlier paper [6] that the single row routing with minimum wire length is $\mathcal{NP}$-complete in the dogleg-free Manhattan model. Now, we shall prove that the previous theorem remains true without the restriction to the dogleg-free case.

In the context of the channel routing, the *density* (or congestion) of a vertical line is the number of nets crossing the vertical line. The *density* (or congestion) of the channel routing problem is the maximum density of all vertical lines through the channel.

We shall call the terminals *left, right* and *middle* according to their relative position to other terminals of the net. The left and right terminals are the *boundary* terminals. We shall call the nets *starting, ending* or *continuing* if it has left, right or middle terminal in the examined section, respectively.

Let us introduce the zone representation. Horizontal segments of any two nets crossing the same vertical line must not be placed in the same track. Let $S(l)$ be the set of nets crossing vertical line l. The *zones* are formed by the different maximal sets. The *zone representation* is popular because it gives a compact description of the problem. We shall examine it in Section 2.3.

2. Definitions and Observations

In this section we introduce some further definitions and observations which facilitate the description and the check of the construction in the proof of NP-completeness. We found interval representation and especially the zone representation very useful for the proof of NP-completeness of the routing problem. For this reason, we shall emphasise in this section how the interval representation and zone representation relate to the original routing problem.

2.1. Routing Problem

In this paper, the only routing model we deal with is the Manhattan model allowing doglegs. For this reason, the name of the routing model is omitted and the Manhattan model is assumed throughout the paper.

Let us call a vertical line *saturated* if its density equals the number of tracks, that is, each horizontal segment is occupied by a wire. Each vertical line is saturated in a *saturated section*. Between two saturated sections a *gap* can be found. Let the gap include its boundary columns as well. Let us call the routing problem *saturated* if each section with maximal density is saturated. In this case, the first part of the gap contains only ending and continuing nets while the other part consists of starting and continuing nets only.

Usually, doglegs can be found at any column of the routing problem. Due to the following trivial lemma, doglegs can occur only in the gaps.

LEMMA 2.1 *No dogleg is possible in a saturated section.*

Proof. A dogleg occupies two horizontal segments at the column where the net changes track and this is not possible at a saturated column.

Let us call the horizontal segments within the saturated sections as *main horizontal segments*. It is unique for a net in a saturated section according to the lemma.

Let us call the vertical wire segments connecting terminals to the horizontal segments as *terminal segments*. We call a dogleg *cheap* if it uses a terminal segment to connect two horizontal segments of a net. Otherwise, if the dogleg uses extra vertical segment, we speak about *expensive* dogleg. According to this, the total vertical wire segments length has two components: one of them is the length of the terminal segments, the other one is the length of the vertical segments of the expensive doglegs. We may assume that the tracks are numbered from 1 to the width. In this case, the vertical wire length of an expensive dogleg is equal to the difference of the tracks connected by the dogleg.

In the dogleg-free case, the position of the middle terminals along the horizontal segment was arbitrary. Now, this is not the case. The possibility of a dogleg and the length of the wire depend on the positions of the middle terminals as well. Let us call a routing problem *nice* if it is saturated and each middle terminal can be found in saturated sections. In this case the possibility of doglegs is restricted. For simplicity, we shall use a nice problem in the proof of NP-completeness.

LEMMA 2.2 *A boundary column of a gap can never be used for doglegs in a saturated problem.*

Proof. A dogleg needs two tracks at his column. There is no space for the additional track in the boundary column, since one track is occupied by the ending or starting net at this column and each of the others is occupied by the horizontal segments of the crossing wires.

If a wire changes from a track to another one, we say that dogleg can be found in the track.

LEMMA 2.3 *No dogleg can be found in the lowest track in case of dense problem.*

Proof. There is no space for the vertical segment of the dogleg because of the vertical segment coming from the terminal.

LEMMA 2.4 *In case of dense problem, doglegs are possible in the second track only in columns where ending or starting net can be found in the first track.*

Proof. If the first track contains a crossing wire, then the terminal segment occupies at least the lowest two tracks and there is no space for the vertical segment of the dogleg.

We shall use further representations of the routing problem in order to examine it.

2.2. *Interval Placement Problem*

If we omit the vertical wire segments from the solution and concentrate only on the horizontal segments we get the *interval placement problem*. This problem is the following. Given a finite set of intervals on a line and w rows. Each interval has to be placed into one of the rows in such a way that two intervals can be placed into the same row if and only if they have no common point. *Fig. 2* depicts an instance of the interval placement problem. This figure not only gives the specification of the problem but also shows one of the solutions.

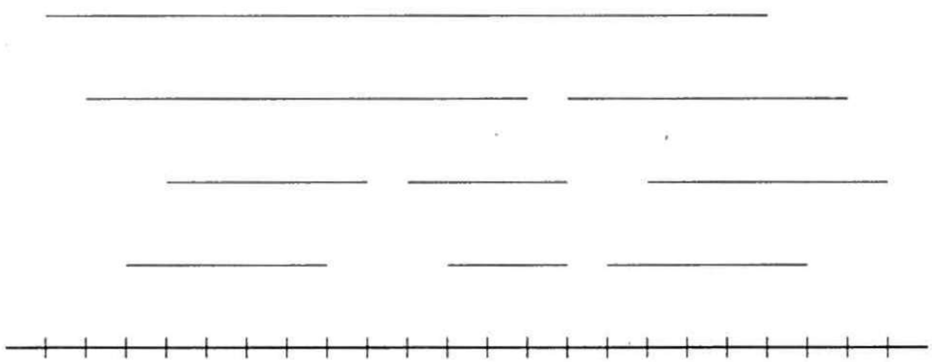

Fig. 2. Interval representation of the routing problem in *Fig. 1*

We often need an optimum solution. The minimum width, that is the minimum number of necessary rows, can be found in linear time ([1], see also [5]. Let us introduce a value to the solutions of an interval placement problem in order to express the vertical wire length. First, we define the value of a solution and we explain only later how it corresponds to the wire length. A weight is assigned to each interval. The interval j has the weight l_j and in a solution it is placed into row r_j. The value of an interval is $r_j \cdot l_j$. The value (v) of a solution of the interval placement problem is the sum of the values of all intervals

$$v = \sum r_j \cdot l_j. \tag{1}$$

The minimum value interval placement problem is as follows: Is there a solution of the interval placement problem for which the value is at most k?

Similarly to the routing problem, we introduce the term saturated. We call an interval placement problem *saturated* if each point except the boundaries of intervals is inside either none of the intervals or exactly w intervals. Thus the intervals have to be placed continuously, without empty place in each row. So two intervals can be placed into the same row if and only if at the end of one of them is the starting point of the other or the section between them can be filled up with other intervals without empty place.

The interval representation is very useful in the Manhattan routing. We can use it for the description of the problem, in this case each net can be given by an interval. The interval representation can describe the solution as well when the intervals represent the horizontal segments of the wires. While in the dogleg-free case, a single interval corresponds to each net in both applications of the interval representation, the interval system corresponding to the solution can be much more complicated in the dogleg case, because the interval belonging to a net is partitioned in the solution according to the places of doglegs.

The weight of an interval shows the number of the terminals belonging to the corresponding net in the dogleg-free Manhattan model. It implies that the interval placement problem represents a routing problem if the weights are integers greater than 1. If the interval placement problem represents a solution of a routing problem allowing doglegs, the weight of an interval gives the number of terminals connected to the horizontal wire segment represented by the interval. To avoid enumerating a terminal more than once, if a terminal segment connects more than one horizontal segment, it is counted only at the horizontal segment placed into the highest row.

The intervals in the interval representation of the solution can be partitioned into smaller ones according to the doglegs. We shall introduce the name *interval switch* for the coinciding ending and starting point of two intervals in the solution if they originate from one interval and they are in different rows. The difference of the two row numbers is the *size of the interval switch.* The doglegs are represented by interval switches in the interval representation. However, the other direction does not work always, that is, an interval switch cannot be always translated into a dogleg in the routing problem because of the overlap of the vertical segments. We shall call an interval switch *legal* if it is realizable in the routing problem as well. Only legal interval switches are allowed in our construction.

2.3. *Zone Representation*

Let us return to the definition of the zone representation to observe the relationship of the elements in the routing problem and in the zone representation. In the zone representation, intervals correspond to the nets and the zone representation can be regarded as an interval placement problem. Thus we can speak about the value of a realization of the zone representation. In the proof of NP-completeness, we shall use the interval placement problem of the intervals in the zone representation. Similarly to the proof in the dogleg-free case, we shall apply saturated problem. A routing problem is saturated if and only if its zone representation is saturated.

The column i belongs to a zone if the nets crossing it together with the net having terminal there is a subset of the set of nets forming the zone. Each zone starts with a left terminal and ends with a right terminal. In case of a saturated routing problem if we compress each saturated section to the unique length and we omit the gaps we get the zone representation. In this case, the gaps are reduced into the single line of the *zone boundaries* and only the main horizontal segments are represented by the intervals of the zone representation.

The *zone weight* is the proportion of the weight of an interval falling into a zone. It shows the number of the terminals of the corresponding net within the zone.

The zone representation can be regarded as a special compact version of the interval representation of the routing problem. In the dogleg-free case, the intervals represent the nets and the horizontal segments of the wires as well. If doglegs are allowed, the zone representation has to be modified if we want to apply it to the description of a solution because doglegs can partition the horizontal segment of a net in the realization.

Fig. 3 depicts the zone representation. A rectangle corresponds to an interval. There are w rows and they have width in contrast with the notation in *Fig. 2* where they are represented by a single line. The rectangle is placed into the row assigned to the interval. The numbers written in the rectangles denote the zone weights or interval weights. We will use this notation, because it is clear and it is suitable for the description of the solution as well as the specification of the interval placement problem. The figure depicts an extension of this notation for the dogleg case, where the doglegs are denoted by oblique lines.

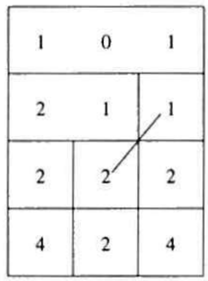

Fig. 3. Zone representation of the routing problem in *Fig. 1*

The possible places of the legal interval switches are drastically reduced in the saturated case according to the following lemma.

LEMMA 2.5 *In case of a saturated problem, legal interval switches are possible only at zone boundaries.*

Proof. Simple application of Lemma 2.1 to the zone representation.

The lemma gives the reason to modify the zone representation by partitioning each interval at the zone boundaries. Let us call the result of this modification as *fragmented zone representation*.

LEMMA 2.6 *If the interval placement problem represents a dense routing problem, no legal interval switch can be found in the lowest row.*

Proof. Simple application of Lemma 2.3 to the zone representation.

LEMMA 2.7 *If the interval placement problem represents a dense routing problem, legal interval switches are possible in the second row only at zone boundaries where the interval endpoint can be found in the first row.*

Proof. Simple application of Lemma 2.4 to the zone representation.

LEMMA 2.8 *The total vertical wire segment length of a nice routing problem is at least the sum of the value of the zone representation of the solution and the sizes of the interval switches.*

Proof. According to the definition, the zone weight (l_j) of an interval shows the number of the terminals of the corresponding net within the zone.

Let us introduce some notations regarding the wire running from the hth terminal to the main horizontal segment.

r_h : the row number of the main horizontal segment.
t'_h : the number of vertical segments on the path concerned.
$k_h^{i'}$: the length of the ith vertical wire segment on the path.
k' : the total vertical length of the paths from each terminal to the main horizontal segments.

The terminal can be either in a saturated section or in a gap. If the terminal is in the saturated section, the row number is equal to the length of the terminal segment starting from it ($t'_h = 1, k_h^{1'} = r_h$). In case of gaps, the length of the terminal segment is unknown because here doglegs can partition the wire running from the terminal to the main horizontal segment into the sequence of some vertical and horizontal segments. However, the sum of the lengths of these vertical wire segments is at least the row number of the main horizontal segment: $\sum_{i=1}^{t'_h} k_h^{i'} \geq r_h$.

Thus, at least so long vertical wire segments belong to each terminal as the row number (r'_j). Furthermore, these vertical segments are disjoint for each terminal if the problem is nice. For this reason, the value of the zone representation, that is the sum of products of the row numbers and weights, gives the total length of the vertical segments mentioned before: $v = \sum r_j \cdot l_j \leq k'$.

This calculation omits the vertical wire segments which connect two neighbouring main horizontal segments of the same net. In this case, interval switches appear in the zone representation of the solution.

We may assume that the main horizontal segment pairs to be connected are numbered. In this case we can apply the notation as follows:

$t''_h, k_h^{i''}, k''$: the same meaning as before but now they refer to the wire connecting the hth main horizontal segment pair.

d_h: the size of the interval switch, that is, the difference of the row numbers of the main horizontal segments.

In this case the vertical length of the path connecting the hth main horizontal segment pair is at least the size of the interval switch: $\sum_{i=1}^{t''_h} k_h^{i''} \geq d_h$. Summarizing this expression for each pair to be connected, we get that the total length k'' of the vertical wire segments connecting main horizontal segments is at least the sum of the sizes of the interval switches.

In case of nice problem instance, the wires running from terminals to the main horizontal segments and the wires connecting two main horizontal segments are disjoint. $k = k' + k'' \geq v + \sum d_h$.

Remark 1 The minimum value of the fragmented zone representation can be easily found by assigning the intervals to the rows according to the decreasing order of their zone weights. It can be used as a lower bound for the total vertical wire segment length.

Remark 2 Somebody could think, that if finding the minimum wire length in the dogleg-free Manhattan model is $\mathcal{NP}$-complete then this would imply directly the $\mathcal{NP}$-completeness of Manhattan routing allowing doglegs. However, the fragmented zone representation is a good example that the routing problem may become solvable in polynomial time after omitting restrictions for the solution because the fragmented zone representation can be regarded as a generalization of Manhattan routing where the horizontal segments can freely change rows at the zone boundaries.

3. Construction

We did not found a trivial way to reduce the dogleg-free case to the general one. We will reduce the satisfiability problem of Boolean formulas (SAT; see GAREY at JOHNSON, 1978) directly to the routing problem with minimum wire length in the Manhattan model in order to show the $\mathcal{NP}$-completeness of the later problem. This reduction is analogous to the one in the dogleg-free case. However, we cannot neglect the positions of the middle terminals between the end points of a net or the relative positions of consecutive right or left terminals, as we did in the former paper. For this reason, we have to examine the interval placement problem together with the original routing problem in contrast to the dogleg-free case where the SAT problem was reduced first to the interval placement problem and then the interval placement problem was reduced to the routing problem.

During the construction we shall use the zone representation for the description of the problem and of the solution. The elements will be the variants of the ones applied in the dogleg-free case. We shall apply a nice routing problem, that is, saturated instance where each middle terminal is placed into the saturated segment, because it simplifies the routing problem.

3.1. Occurrence of a Variable

Fig. 4 shows the part of the construction corresponding to an occurrence of a variable. Let us call it as the *variable-occurrence element*. As we remarked after Lemma 2.8, the fragmented zone representation is useful to find a lower bound for the total length of the vertical wire segments. The minimum value of the fragmented zone representation is 74, it is less by one than the minimum value solutions without interval switches.

The vertical wire segment length is greater than the value at least by the sizes of the interval switches. It means that using interval switches we cannot decrease further the minimum total vertical wire segment length of the variable-occurrence element but it is still possible to find another solution whose vertical wire segment lengths are the same as the minimum one in the dogleg-free case. Therefore, the sizes of interval switches are one unit long altogether if the total vertical wire length

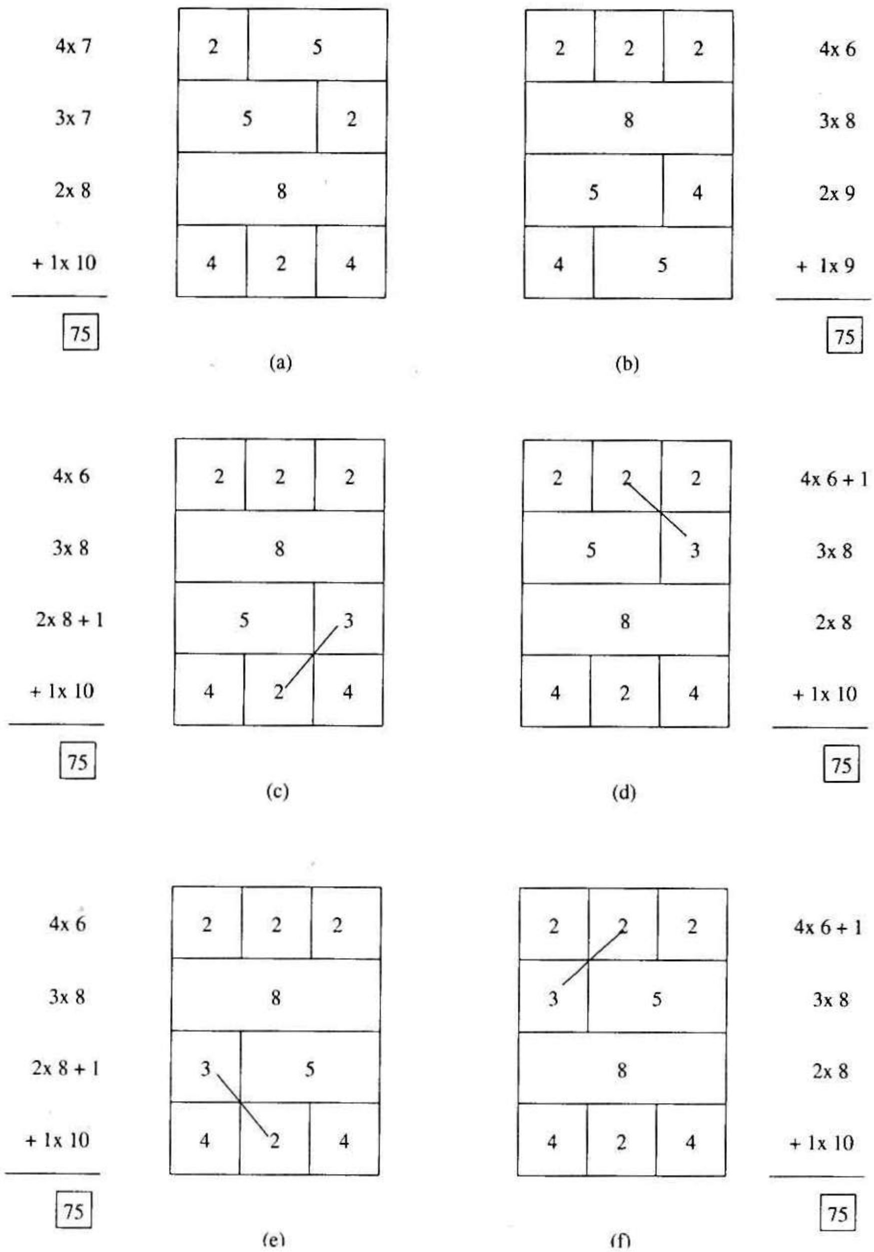

Fig. 4. The minimum length solutions of the element corresponding to an occurrence of a variable

r0	2	5		r1	2		5	r2
r0	5		2	r1	5		2	r2
r0		4		r1		2		r2
r0	4	2	4	r1	4	2	4	r2

Fig. 5. An element corresponding to a variable

is minimum, that is, only one interval switch is possible between two neighbouring rows. *Fig. 4* depicts all minimum length solutions. If the rows in a figure are permuted, we do not consider the new solution to be different. In the dogleg-free case, we distinguished two different minimum value realizations. Now, we have more, for this reason we speak about types of realizations. Let us call the minimum value realizations of type A and $\overline{A}$, if the unifying interval occupies the second and the third rows, respectively. Let the realizations *Figs. 4a, 4d, 4f* (A) correspond to true, and *Figs. 4b, 4c, 4e* $(\overline{A})$ correspond to false.

3.2. Variable

The *variable element* occupies four adjacent rows. They contain as many variable-occurrence elements as the number of occurrences of the variable in clauses. The variable in *Fig. 5* occurs in two clauses. The intervals within the variable element are the *variable intervals*. *Variable-connecting intervals* fill up the space in the rows between two adjacent variable-occurrence elements and at the end and beginning of the rows. Their weights are denoted by r_i in the figure. The zone weights of the variable-connecting intervals are equal to each other within the same zone. These weights will be calculated in Section 3.4. All the intervals with weight 8 belonging to the same variable and the variable-connecting intervals between them are merged into one long interval. As a result, we get one long interval covering the whole length of the variable element. Let us call it *unified interval.* We will see that it ensures that the minimum value solution of the variable element uses the same type of realization – either A or $\overline{A}$ – at each variable-occurrence element belonging to it. The main difference from the proof for dogleg-free case is that not only the interval with weight 8 but the intervals with weight 4 and 5 lying at the right side of the variable-connecting intervals are merged with the coinciding variable-connecting intervals.

The intervals with weight 4 on the left side, with weight 2 on the right side and with weight 2 in the middle of the variable-occurrence element will be used for exchanges with clause intervals, see the next section. Let they be called as *variable exchange intervals.*

LEMMA 3.1 *The total vertical wire length of the variable element is minimum if and only if either realization of type A or $\overline{A}$ occurs at each element corresponding to an occurrence of the same variable.*

Proof. The realizations of type A and $\overline{A}$ of a variable-occurrence element are the minimum length realizations. Since the zone weights of the variable-connecting intervals are equal to each other within the same zone, the value of the placement cannot be decreased either on the segment between the variable-occurrence elements. For this reason, if the unified interval or the variable-connecting intervals would change from one row to another then the wire length would be longer than in the dogleg-free case because of the size of the interval switch (Lemma 2.8). Consequently, if the variable element has the minimum value, then the unified interval occupies the same row and it forces each element belonging to the same variable to have the same type.

3.3. Clause

The element corresponding to a clause is called *clause element* and it is similar to the one in the dogleg-free case. A specific example of a clause element is shown in *Fig. 6*. The lower two rows are the *clause rows*, they contain the *clause intervals* and the rows above them are the *variable rows*. *Clause-connecting intervals* start and finish the clause elements. The clause intervals B and $\overline{B}$ are called as *clause exchange intervals*. The clause intervals other than the clause-connecting and the exchange ones are the *literal-connecting intervals*. Two overlapping literal-connecting intervals always form a pair of a shorter and a longer interval. Their weights are denoted by s_i and l_i, respectively. The weights of the clause connecting intervals are denoted by t_1 and t_2 in *Fig. 6*.

The zones between the variable-occurrence elements are the *constraining zones*. The constrain zones are denoted by striped columns in the figure. We speak about *constrained solution* if the clause intervals run in the clause rows in the constraining zones. We ensure this by choosing a great value as zone weight for the literal-connecting intervals in the constraining zone. Let this weight be called as *constraining weight* denoted by c and it will be determined in Section 3.5.

The zone weights of the clause intervals are usually equal to the maximum weights within the variable rows in their zone. The right zone of $\overline{B}$ and the constraining zone belong to the exceptions. Furthermore in the rightmost zone of the longer clause-connecting interval, its zone weight is greater by 2 than the one of the other clause interval there. The zone weight of the shorter literal-connecting interval is greater by one than the one of the other clause interval in the last zone within the clause. See *Fig. 7* for an example of the zone weights of a clause element.

We proved in the dogleg-free case that only the exchange intervals could be put into a variable row. We can prove the same fact in the dogleg case provided that the solution is constrained.

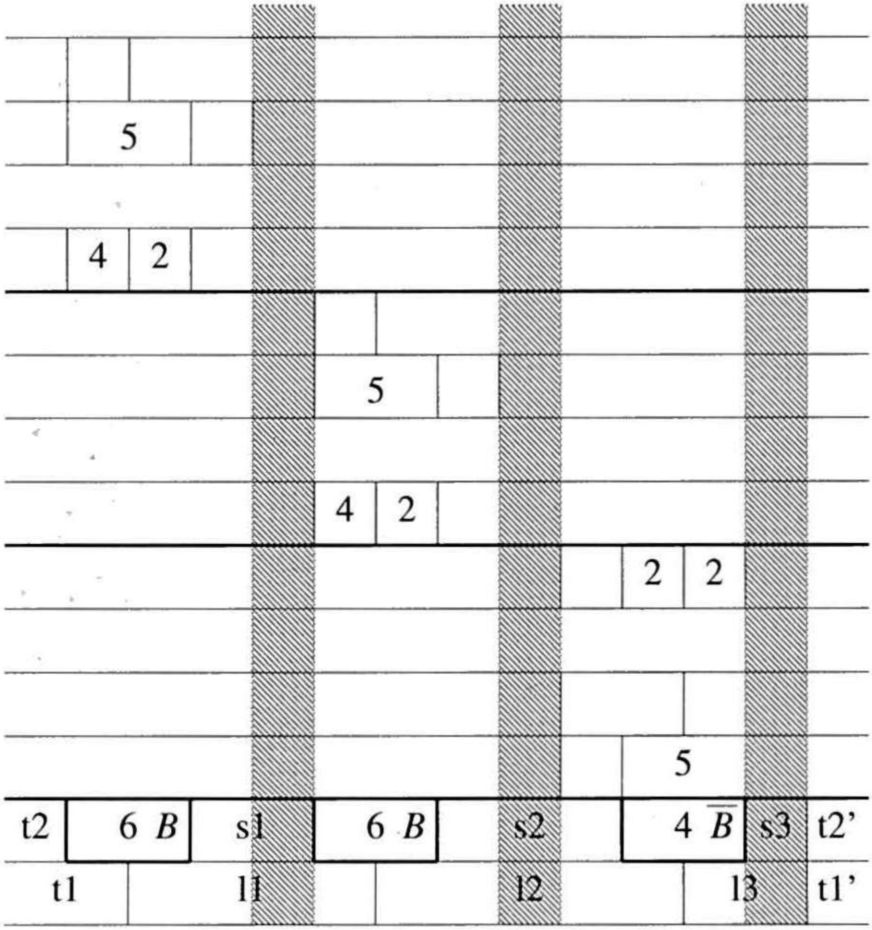

Fig. 6. An element corresponding to the clause $x_3 \vee x_2 \vee \overline{x}_1$

2	4	2	4	c	4	2	4	c	4	2	2	c+1	2
2	6	2	4	c	4	2	4	c	4	2	2	c	2

Fig. 7. The zone weights of the clause element in *Fig.6*

LEMMA 3.2 *If the solution is constrained and the clause intervals have no interval switches then only the exchange intervals can be placed into a higher row.*

Proof. Without interval switches, the literal-connecting intervals are fixed to the clause rows in the constrained solution. The clause-connecting intervals are also in the clause rows, otherwise variable intervals would fill their place in the clause rows, but it is impossible without interval switches. Consequently, only the clause exchange intervals can be placed into variable rows from the clause intervals.

LEMMA 3.3 *Let us assume that the solution is constrained, there are no interval switches in the clause rows and the variable-occurrence elements together with the variable-connecting intervals belonging to the same variable are placed into adjacent rows. Then the placement of intervals belonging to them has the minimum length if and only if realization of type A belongs to interval B or realization of type $\overline{A}$ belongs to $\overline{B}$ at least at one occurrence of a variable at each clause.*

Proof. We proved in the previous lemma that only the clause exchange intervals could be put into the variable rows. If the clause exchange intervals remain in the clause rows, the value of the clause-element is greater by one than the minimum value of its fragmented zone representation.

Let us consider the minimum value solution if an exchange interval is placed into a variable row. Since there are no interval switches, the place of interval $\overline{B}$ can be occupied only by the variable exchange intervals. The other intervals were merged with the variable-connecting intervals in the variable element in order to avoid the changes with them. Interval B can be exchanged with intervals other than the variable exchange intervals as well, but this is not worth doing because the total value increases.

The change of the variable and clause exchange intervals enables the exchange of the two rows of the clause element on a section, which reduces the total value by one. Let us call this exchange of clause rows an *improving exchange*. It does not matter whether one or more intervals belonging to the same clause are exchanged with intervals of variable rows because the total value can be reduced by one in each case.

We may assume, that the realization of the variable-occurrence element belongs to one of the types of the minimum length realizations. Let us note in this case, that intervals B and $\overline{B}$ can be exchanged with variable exchange intervals only if the realization of the variable-occurrence element is of type A and $\overline{A}$, respectively. The value of the clause element can be reduced by one if and only if the corresponding realization of the variable-occurrence element and clause exchange interval can be found at one or more occurrences of variables. The total value is minimum if this holds at each clause. Since there are no interval switches in the clause rows, the minimum value of the zone representation means minimum wire length as well.

At this point, we cannot eliminate the position of terminals in the construction any more. We assign terminals to the shorter literal-connecting intervals so that their left terminals are the last among the left terminals in the same gap. This assignment will help to avoid doglegs.

Let us consider the zone representation of the minimum length dogleg-free solution (*Fig. 8*). The only zone, where the value can be reduced in the clause rows, can be found at the right half of the exchange interval $\overline{B}$. Let us call this zone as

0	0	0	0	0	0	0	0	0	0	0	0	0	
1	2	2	3	1	0	0	0	0	0	0	0	0	
1	3	2	2	1	0	0	0	0	0	0	0	0	
1	3	2	3	1	0	0	0	0	0	0	0	0	
1	4	2	4	1	0	0	0	0	0	0	0	0	
1	4	2	4	1	2	2	2	1	0	0	0	0	
1	4	2	4	1	3	2	3	1	0	0	0	0	
1	4	2	4	1	3	2	3	1	0	0	0	0	
1	4	2	4	1	4	2	4	1	0	0	0	0	
1	4	2	4	1	4	2	4	1	2	2	2	1	
1	4	2	4	1	4	2	4	1	3	2	3	1	
1	4	2	4	1	4	2	4	1	3	2	4	1	
1	4	2	4	1	4	2	4	1	4	2	3	1	
2	4	2	4	c	4	2	4	c	4	2	2	c	2
2	6	2	4	c	4	2	4	c	4	2	2	c+1	2

Fig. 8. The minimum value solution of the clause element in *Fig. 6*

conflict zone. Here greater weights can be found in the variable rows than in the clause rows. In any other zone the minimum is ensured by the decreasing order of the zone weights.

LEMMA 3.4 *If the solution is constrained then clause or variable exchange intervals occupy the clause rows in the conflict zone.*

Proof. The proof is indirect, let us examine that a variable interval other than the variable exchange intervals occupies one of the clause rows in the conflict zone. In this case, it has to return to a variable row before the constraining zone. It is not possible in the first row, because there is no legal interval switch (Lemma 2.6). If the longer literal-connecting interval occupies the first row, the variable interval could not leave the second row before the constraining zone because no interval endpoint can be found in the first row there and in this case no legal interval switch is possible there in the second row according to Lemma 2.7. Consequently, the longer literal-connecting interval and a variable interval occupies the second row in the constraining zone and in the conflict zone, respectively. Both of these intervals have to switch row because they overlap one another. The only possible place for the switch is the boundary between the conflict zone and the constraining zone.

Now, we have to examine the routing problem behind the zone representation. The legal interval switches represent doglegs in the routing problem. We chose the rightmost column of the gap for the left terminal of the shorter literal-connecting interval and this column is unusable for dogleg (Lemma 2.2). Since the problem is dense, only the column of the right terminal of the net ending in the first row remains in the current gap where the interval could switch from the second row into another. However, one column is not enough for both wires represented by the longer literal-connecting interval and the variable interval to switch into another row.

3.4. *Boolean Formula*

We still have to complete the construction determining the first and last zones of the construction. Let each variable row be introduced by variable-connecting intervals and the clause rows started by intervals with constraining weights and clause-connecting intervals before the first clause element. Let the clause rows be finished by clause-connecting intervals.

We should calculate the weight of the variable-connecting intervals. Let us order the variables from 1 to n and let the variable elements be placed above the clause rows according this order. The zone weight of the variable-connecting intervals are the maximum of the weights above them within the same zone. Let us formulate this. Each zone either crosses a variable-occurrence element or neighbouring to one or two variable-occurrence elements. We introduce the notations as follows:

z^j: The index of the variable or the highest index if there are more variables whose variable-occurrence element is crossed by or neighbouring to the zone j.

p^j: It is an attribute characterising the position of the zone relative to the variable-occurrence element concerned. Three different relative positions are distinguished, namely boundary, middle, neighbouring.

l_i^j: The zone weight of the interval-connecting interval belonging to the ith variable in the jth zone.

$$l_i^j = \begin{cases} 4 & \text{if } p^j \text{ is boundary and } z^j \geq i, \\ 2 & \text{if } p^j \text{ is middle and } z^j \geq i, \\ 1 & \text{if } p^j \text{ is neighbouring and } z^j \geq i, \\ 0 & \text{if } z^j < i. \end{cases}$$

We filled in the zone weights of the variable-connecting intervals in *Fig. 8* according to this.

Now, we can prove that the assumption of the missing interval switches is correct.

LEMMA 3.5 *The minimum length solution does not use legal interval switches other than the ones within the variable-occurrence elements already introduced.*

Proof. Let us examine whether the value of the zone representation can be reduced using legal interval switches. There are only two locations where the value of the zone representation can be improved, in the clause rows at the conflict zone and within the variable-occurrence elements. (These areas are denoted by dashed boxes in the *Fig. 8.*) According to Lemma 3.4 the value of the conflict zone cannot be reduced, because no variable interval can be placed there into the clause rows.

Now, let us focus on the variable-occurrence elements. We studied them in Section 3.1 but we have to examine whether there are possibilities of improvement using the intervals from the other rows. In order to decrease the value, we can try to order the intervals within the zones, to exchange intervals with shorter ones, or to introduce one-unit length interval switches. After short examinations, we can conclude, that the value of a variable-occurrence element cannot be decreased except by the interval switches already introduced at the variable-occurrence elements (*Fig.4*).

Since we failed in reduction of the value of the zone representation, no more interval switches are possible in the minimum length solution according to Lemma 2.8.

LEMMA 3.6 *If the solution is constrained, and there is no interval switch between the rows belonging to different variables, then the variable-occurrence elements and the variable-connecting intervals belonging to the ith variable are in the rows $4i-1, 4i, 4i+1, 4i+2$ in case of the minimum length solution.*

Proof. The rows are filled with intervals without empty segments in case of the saturated zone representation, so the consecutive intervals within the same row have common boundaries. Since none of the variable intervals belonging to different variables has common boundary, a variable interval is placed into a row containing interval belonging to the same variable or, according to Lemma 3.2, clause exchange intervals. Consequently, if one of the variable intervals is put into a row belonging to another variable, then the whole row is exchanged.

Originally, the intervals belonging to the variable i occupy the corresponding variable rows, that is the rows $4i-1$, $4i$, $4i+1$, $4i+2$. Let l'_i denote the total weight of a row belonging to the ith variable. $l'_i > l'_j$ if $i < j$ according to the weight assignment of the variable-connecting intervals. Therefore the reordering of the rows would increase the value of the placement and the wire length of the routing.

We remark that according to Lemmas 3.2 and 3.3, variable exchange intervals can be replaced by clause exchange intervals and they can be put into the clause rows in the minimum length solution.

3.5. The Weights of the Literal-Connecting Intervals

We still have to determine the zone weights of the literal-connecting intervals within the constraining zones, the constraining weights.

LEMMA 3.7 *If the weights of the literal-connecting intervals in the constraining zone are at least* $(16n + 1)o + 2$ *then these intervals are placed into the lowest two rows in the minimum length solution. (n is the number of the variables and o is the number of the variable occurrences in the Boolean formula.)*

Proof. Let the value of the fragmented zone representation be v_f, the minimum wire length of the solution described in the construction be l_c and the minimum wire length be l_r if one of the literal-connecting intervals is placed into the third or higher row in the constraining zone.

We can calculate easily an upper bound for the difference of l_c and v_f. The vertical wire length is greater by one at each variable-occurrence element in our construction than the minimum value of its fragmented zone representation. The other location where the value of the construction could be improved is the conflict zone where the clause intervals with zone weight 2 occupy the first two rows under the variable intervals whose zone weights are at most 4. This results in an upper bound $16n$ for the difference of the value of the conflict zone in the construction and the minimum value of the fragmented zone representation. Summarizing the differences, we get $v_f \leq l_c \leq v_f + (16n + 1)o$.

If the zone weights of the literal-connecting intervals are $(16n + 1)o + 2$ in the constraining zone and the value of the zone is minimum, then the first two rows are occupied by them, and intervals with zone weight 1 or 0 are placed into the rows above them. If one of the literal connecting intervals is placed into a higher row, then the wire length will be greater than the original value at least by the zone weight of the literal-connecting interval minus the zone weight of the variable-connecting interval replaced with it. $l_r \geq v_f + (16n + 1)o + 1$.

Consequently, $l_c \leq l_r$, that is, the minimum length solution contains the literal-connecting intervals in the clause rows in the constraining zone.

3.6. Minimum Wire Length and $\mathcal{NP}$-Completeness

We have shown a construction how to assign a routing problem to each instance of SAT. We have to calculate an appropriate value for k which is contained by the instance of the minimum wire length Manhattan routing problem as well and it is equal to the minimum wire length which can be achieved if and only if improving exchange can be realized at each clause. Details are omitted, but it is necessary to note that k can be determined in polynomial time, because each weight and the value of each component of the optimum solution can be calculated in polynomial time.

THEOREM 1 *Single row routing with minimum wire length is $\mathcal{NP}$-complete in the Manhattan model if the width is equal to the minimum width.*

Proof. We may assume that the length of the horizontal segments of a wire is equal to the difference of the position of the left terminal and the right terminal of the corresponding net. We can do this, because introducing wires crossing the same vertical line more than once does not cause decrease in the length of the vertical wire segments.

The total length of the horizontal segments is the same in any realization of the routing instance. Thus the minimization of the wire length is equivalent to the minimization of the vertical segment length.

The proof is similar to the dogleg-free case. Each element of the construction can be determined in polynomial time in the size of the Boolean formula, and the whole construction applies polynomial number of building elements. We can conclude that each instance of SAT can be translated into an instance of the minimum wire length Manhattan routing problem in polynomial time.

We shall prove that the interval placement problem has a solution with total wire length k if and only if the Boolean formula is satisfiable. By Lemmas 3.7 and 3.5, the solution is constrained, and there is no interval switch other than the ones within the variable-occurrence elements. In this case, Lemma 3.2 guarantees that the literal-connecting and clause-connecting intervals are placed into the lowest two rows in the minimum length solution. By Lemma 3.6, the variable-occurrence elements and the variable-connecting intervals are placed into the corresponding variable row. By Lemma 3.1, the same realization belongs to each element corresponding to the occurrence of the same variable in the minimum length solution. By Lemma 3.3, the solution of the interval placement problem has the least length if and only if realization of type A belongs to interval B or realization of type $\overline{A}$ belongs to interval $\overline{B}$ at least at one occurrence of a variable in each clause element.

Suppose first that the problem can be routed with length k. If the variable element has a realization A or $\overline{A}$, let the corresponding Boolean variable true or false, respectively. In this case each clause and the entire Boolean formula is satisfiable. Conversely, if the Boolean formula satisfiable, apply realization A or $\overline{A}$ to the variable element depending on the value of the Boolean variable. In this case we can exchange variable and clause exchange intervals in each clause element and we can achieve the wire length k.

We have assumed so far that the number w of the rows is given in the problem and it was equal to the minimum width. However, the total wire length may be shorter if more rows are used than necessary.

THEOREM 2 *The minimum total wire length Manhattan routing problem with arbitrary width is $\mathcal{NP}$-complete.*

Proof. The proof is the same as in the dogleg-free case (Lemma 7 in paper of SZKALICZKI, [?]). The essence of the proof is the reduction of the minimum wire length Manhattan routing problem with minimum width to the case when the width is arbitrary by introducing two-terminal nets whose horizontal segments cover the whole original problem. If the number of the inserted nets is high enough, the wire running in the highest row can be longer than the total vertical wire length of the original problem. Consequently, if the width of the channel increases, then the wire length will be longer than the minimum wire length with minimum width.

4. Conclusions

The routing problem is known to be $\mathcal{NP}$-complete in many cases. Routing with minimum wire length in the dogleg-free Manhattan model had also been proved to be $\mathcal{NP}$-complete before. The present paper completed this result by showing that finding the minimum wire length is computationally difficult in the Manhattan model even if we do not restrict ourselves to the dogleg-free special case.

References

[1] GALLAI, T. unpublished; announced in A. Hajnal and J. Surányi: Über die Auflösung von Graphen in vollständige Teilgraphen, *Ann. Univ. Sci. Budapest Eötvös*, **1** (1958), pp. 115–123.

[2] GAREY, M. R. – JOHNSON, D. S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, California, 1978.

[3] LAPAUGH, A. S., A Polynomial Time Algorithm for Optimal Routing around a Rectangle, *Proc. 21st FOCS Symp.*, 1980, pp. 282–293.

[4] LENGAUER, TH., *Combinatorial Algorithms for Integrated Circuit Layout*, Teubner, Stuttgart. Wiley, Chichester, 1990, pp. 534–551.

[5] RECSKI, A., Minimax Results and Polynomial Algorithms in VLSI Routing, *Ann. Discrete Math*, **44** (1992), pp. 261–273.

[6] SZKALICZKI, T., Routing with Minimum Wire Length in the Dogleg-free Manhattan Model is NP-complete, *SIAM Journal on Computing*, **29** No. 1 (1999), pp. 274–287.

[7] SZYMANSKI, T., Dogleg Channel Routing is NP-complete, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **4** No. 1 (1985), pp. 31–41.