A Re-router for Reducing Wire Length in Multi-Layer No-Dogleg Channel Routing

Swagata Saha Sau[a*], Rajat Kumar Pal[a]
[a]*Department of Computer Science and Engineering, University of Calcutta, JD - 2, Sector-III, Salt lake City, Kolkata 700 098, West Bengal, India*

Abstract*—The minimization of total wire length is one of the most key issue in VLSI physical design automation, as it reduces the cost of physical wiring required along with the electrical hazards of having long wires in the interconnection, power consumption, and signal propagation delay. So, it is still important as cost as well as high performance issue. The problem of reduced wire length routing solutions in no-dogleg reserved two-layer (VH) and multi-layer (V_iH_i, $2 \le i < d_{max}$ and V_iH_{i+1}, $2 \le i < d_{max} - 1$) channel routing is NP-hard, so, it is interesting to develop heuristic algorithms that compute routing solutions of as minimum total (vertical) wire length as possible. Here we propose two algorithms to reduce the total (vertical) wire length in channel routing problem. First we develop an efficient re-router Further_Reduced_Wire_Length (FRWL) to optimize the wire length in the reserved two-layer (VH) no-dogleg channel routing model and then we develop an algorithm Multi-Layer_Reduced_Wire_Length (MLRWL) to minimize the total (vertical) wire length in channel routing problem in the reserved multi-layer (V_iH_i, $2 \le i < d_{max}$ and V_iH_{i+1}, $2 \le i < d_{max} - 1$) no-dogleg Manhattan routing models, where vertical and horizontal layers of interconnect alternate. Experimental results computed for available benchmark instances indicate that the algorithms perform well.*

Keywords—*Channel routing problem, Manhattan routing, No-dogleg, Parametric difference, Wire length minimization, VLSI.*

I. INTRODUCTION

The minimization of area in channel routing problem (CRP) is important for cost optimization point of view as well as for reducing impurities present on a wafer. The research in this field almost is saturated in literature [1,5-7,9,10,14]. We know that the wire length minimization problem in channel routing is also a very significant crisis but this problem has not been well conscious so intensely by the researchers for any cause behind. This is also a significant problem as minimization of wire length also reduces routing cost, but the problem is more imperative as a high performance factor of computing a routing solution. More wire length means more delay and also more congestion of wire segments that may produce more heat and cause for signal interference.

Thus, channel routing with minimum total wire length not only reduces cost of the physical wire for interconnection in circuit but also reduces delay of propagating signals, power consumption and that gives better electrical performance of the chip. The computation of reduced wire length routing solutions in no-dogleg reserved two-layer (VH) and multi-layer (V_iH_i, $2 \le i < d_{max}$ and V_iH_{i+1}, $2 \le i < d_{max} - 1$) channel routing is NP-hard problem [9,13]. In literature, only a very few algorithm have been devised for minimizing total wire length of the channel [2,3,8,9,11,12]. Most of existing routing solutions of the problem stated above introduced many doglegs which often removes the cyclic vertical constraints by
sacrificing more via in the channel. From electronic circuit point of view, each dogleg introduces one or more additional vias for each net, which increases delay, electrical hazards, cost of design, and often wire length whereas, decreases performance of the circuit. Thus, usually a no-dogleg channel router is definitely acceptable for increasing the performance of a chip to be designed.

A. Basic Definitions

Generally, a channel is a rectangular routing region between circuit blocks having two parallel rows of fixed terminals on a chip floor [1-3,5-14]. Terminals (pins) to be connected inside a channel are placed on the periphery of the blocks. A *net* is a collection of terminals (assigned a single number) and they are connected together by electrical wires. If a net contains only two (more than two) terminals, is called a *two-terminal net* (*multi-terminal net*). Vacant terminals are assigned the number zero that not to be connected. Usually, a channel is represented by *channel specification* (or *net list*) that contains two vectors of equal length of net numbers (including vacant terminals), where these numbers are assumed as fixed terminals along the length of the channel.

The *span* is actually the *interval* of a net that is spread between the leftmost and rightmost columns of the net (along the length of the channel). Other than fixed terminals (along two opposite sides), a channel may have floating *terminals* (along the other two opposite sides of the rectangle, as the row number for

its related net is not fixed before computing a routing solution). The set of nets that enters into the channel from its left (right) end is called *left* (*right*) *connection set* (*LCS* (*RCS*)) in the net list [9,11]. In channel routing, interconnections are made within a rectangular region having no obstructions.

In this paper, we specifically consider the reserved layer no-dogleg Manhattan channel routing model, where a set of layers is assigned for routing only horizontal wire segments and the remaining set of layers is assigned for routing only vertical wire segments of the nets belonging to a channel. In Manhattan routing, we allow only rectilinear wiring for all necessary interconnections. In no-dogleg routing, single horizontal wire is assigned to track for all individual net. This routing model is most practical as it is the simplest and modular routing model adopted in designing a maximum of marketable chips, whose performance is also within the limit of tolerance.

In this paper, we have developed two algorithms for wire length minimization: (a) One for the two-layer VH routing model and (b) the other for the multi-layer (V_iH_i, $2 \le i < d_{max}$ and V_iH_{i+1}, $2 \le i < d_{max} - 1$) channel routing models. Both the algorithms devised are iterative in nature and for each iteration it takes time $O(n+e)$, where n denotes the number of nets present in the channel and e denotes the size of a constrained graph under consideration (that we use to represent the inherent constraints of CRP). As each of the algorithms iterates for t times, the overall time complexity of our algorithms become $O(t(n+e))$, where t denotes the number of tracks required to route a channel.

The two inherent constraints are termed as horizontal constraints and vertical constraints present in the channel. Two nets are said to be horizontally constrained, if their intervals overlap when they are assigned to the same track. Horizontal constraints are represented by an undirected graph, known as *horizontal constraint graph*, HCG (V, E_h) [9-12], where $V = \{v_i \mid v_i$ denotes interval I_i related to net $n_i\}$ and $E_h = \{\{v_i, v_j\} \mid I_i$ and I_j overlap$\}$. The complement of the horizontal constraint graph is *horizontal non-constraint graph* and we call it HNCG [9-12]. The vertical constraint represented by a directed graph known as *vertical constraint graph*, VCG (V, E_v) [9-12], where $V = \{v_i \mid v_i$ represents interval I_i related to net $n_i\}$ and $E_v = \{(v_i, v_j) \mid n_i$ has vertical constraint with $n_j\}$. The number of nets passing through a column is termed as the *local density* of that column. The *density* of the channel, d_{max} is the maximum of all local densities.

B. Organization of the Paper

The paper is organized as follows. In Section II, we briefly state the hardness of CRP of wire length minimization and then formulate the algorithms developed in this paper. The proposed algorithms for optimizing wire length in two- and multi-layer no-dogleg channel routing is discussed in Section III. Experimental results based on an exhaustive experimentation for most of the existing benchmark channel instances are made for different multi-layer channel routing models; all these results have been included in Section IV. The present article is concluded with few remarkable points in Section V.

II. FORMULATION OF THE PROBLEM

In this section, the nature and formulation of wire length minimization problem in channel routing is discussed. Actually, the reserved two-layer (VH) and multi-layer (V_iH_i, $2 \le i < d_{max}$ and V_iH_{i+1}, $2 \le i < d_{max} - 1$) no-dogleg CRP of wire length minimization is NP-hard [9-14]. Thus, the development of heuristic algorithms that compute routing solutions with minimum wire length is one of the probable way outs, which is at the same time truly interesting. In this paper we have developed such polynomial time (heuristic) algorithms to solve the said problem.

Recently a graph based algorithm *Modified_Track_Assignment_Heuristic* (*MTAH*) was developed to reduce total wire length in the reserved two-layer no-dogleg Manhattan routing model [11]. Here authors tried to reduce total wire length by assigning weights to nets. They applied the maximum weighted clique computation algorithm [4] for selecting a set of nets with non-overlapping intervals for assignment to a track in each iteration. After every iteration, the algorithm deletes the nets that have recently been assigned into a track, and assigns the remaining nets on subsequent iterations with similar but smaller instance of the CRP. We considered eight sub-modules to reduce the total wire length of a channel and finally took the optimal one among the routing solutions of these modules. However, even then we observed that there are scopes to reduce the total vertical wire length of these routing solutions further. In this paper we consider such cases as well in addition to several other routing solutions computed by other algorithms existing in literature to work out the same.

One of our objectives in this work is to compute maximally reduced total wire length routing solutions in the reserved two-layer (VH) and multi-layer (V_iH_i, $2 \le i < d_{max}$ and V_iH_{i+1}, $2 \le i < d_{max} - 1$) no-dogleg routing models. The total wire length of a routing solution is the sum of the total horizontal wire length and the total vertical wire length of the routing solution. Here we have considered the reserved layer Manhattan routing model and terminal positions are also fixed of a given

channel specification. Hence, the total horizontal wire length in a channel is constant. Hence, to minimize the total wire length implies to minimize the total vertical wire length of a routing solution. If the total number of top (bottom) terminals TT_i (BT_i) is more than the total number of bottom (top) terminals BT_i (TT_i) of net n_i, then the assignment of the net towards the top (bottom) row reduces the total (vertical) wire length. We call $TT_i - BT_i$ the *parametric difference* of net n_i, and denote it by pd_i [9,11].

We suggest a re-router *Further_Reduced_Wire_Length (FRWL)* that minimizes total wire length in the reserved two-layer (VH) no-dogleg routing model significantly. We develop another algorithm *Multi-Layer_Reduced_Wire_Length (MLRWL)* that also reduces total wire length in the reserved multi-layer (V_iH_i, $2 \le i < d_{max}$ and V_iH_{i+1}, $2 \le i < d_{max} - 1$) Manhattan routing models.

We consider the multi-layer model in such a way that vertical layer (V) and horizontal layer (H) are to be placed alternating. We distribute the nets of the computed two-layer routing solutions using *FRWL* (containing T number of tracks) into multi-layer (V_iH_i, $2 \le i < d_{max}$) routing model in such a way that the nets of the first, the second up to the i^{th} tracks of the two-layer routing solution are placed in the first track of the first, the second up to the i^{th} horizontal layers, respectively, and the vertical wires for interconnection of the nets placed in the first, the second up to the i^{th} horizontal layer are placed in the first, the second up to the i^{th} vertical layer, respectively.

Similarly, for the rest we take i successive tracks at a time and place them to the same track of i different horizontal layers. In this routing solution each horizontal layer has $\lceil T/i \rceil$ number of tracks containing nets except possibly the last few horizontal layers. Here some tracks of horizontal layers may be unfilled. As the first $i - 1$ horizontal layers are flanked by vertical layers, hence the nets placed in these horizontal layers render no vertical constraints. For the ineffectiveness of vertical constraints, the top terminal and bottom terminal of different nets in a column can be joined through wires placing them into separate vertical layers. Then we apply *FRWL* on i pairs of VH layers separately in computing further reduced wire length multi-layer (V_iH_i, $2 \le i < d_{max}$) channel routing solutions.

For multi-layer (V_iH_{i+1}, $2 \le i < d_{max} - 1$) routing model, we distribute the nets of the computed two-layer routing solutions using *FRWL* (containing T number of tracks) in such a way that the nets of the first, the second up to the $(i+1)^{th}$ tracks of the two-layer routing solution are to be placed in the first track of the first, the second up to the $(i+1)^{th}$ horizontal layers, respectively, and the vertical wires for interconnection of the nets placed in the extreme horizontal layers are to be placed in the adjacent vertical layers, and the vertical wires for interconnection of the nets placed in the horizontal layers flanked by vertical layers are to be placed into any one of adjacent vertical layers. Similarly, for the rest we take $(i+1)$ successive tracks at a time and place them to the same track of $(i+1)$ different horizontal layers. In this routing solution each horizontal layer has $\lceil T/(i+1) \rceil$ number of tracks containing nets except possibly the last few horizontal layers. Here some tracks of horizontal layers may be unfilled. As the middle $i - 1$ horizontal layers are flanked by vertical layers, hence, the nets placed in these horizontal layers as such have no vertical constraints.

III. The Proposed Algorithm

We have developed two routing algorithms in this paper. First we design algorithm *Further_Reduced_Wire_Length (FRWL)* as has been depicted in Fig. 1 to compute further reduced total (vertical) wire length routing solutions in the two-layer (VH) no-dogleg routing after reassigning the nets to tracks of the routing solutions that are obtained using algorithm *Modified_Track_Assignment_Heuristic* (*MTAH*) [11]. Our proposed algorithm computes optimal channel routing solutions for minimizing wire length of the channels. In this paper, we have designed another algorithm *Multi-Layer_Reduced_Wire_Length (MLRWL)* as shown in Fig. 2 that minimizes the total wire length in the multi-layer (V_iH_i, $2 \le i < d_{max}$ and V_iH_{i+1}, $2 \le i < d_{max} - 1$) no-dogleg routing models.

Now we analyze the time complexity of our first algorithm *FRWL*. The algorithm has an iterative and a non-iterative part of computation. The time complexity to generate HNCG is $O(n+e)$, and that to generate VCG is $O(n)$ time, where n denotes the number of nets in the given channel and e denotes the size of HNCG. If t is the number of iterations required to route the channel, then the time complexity of the iterative part of the algorithm is $O(t(n+e))$, and hence, the overall time complexity of *FRWL* is also $O(t(n+e))$.

For multi-layer V_iH_{i+1}, $2 \le i < d_{max} - 1$ routing model, the method is very similar to *MLRWL* for computing a V_iH_i, $2 \le i < d_{max}$ routing solution. Here the differences are in Step 3 ($1 \le J \le i+1$ instead of $1 \le J \le i$) and in Step 5 (middle $i - 1$ horizontal layers are flanked by vertical layers instead of the first $i - 1$ horizontal layers).

Algorithm *Further_Reduced_Wire_Length (FRWL)*

Input: A two-layer no-dogleg routing solution of a channel specification using *Modified_Track_Assignment_Heuristic* (*MTAH*) for wire length minimization.

Output: A two-layer routing solution after reconsideration of nets for further reduced wire length.

Step 1: Construct HNCG and VCG.
Topmost_track ← 0
List_A ← NULL
List_B ← NULL
Step 2: Construct a list *Net_List* containing the nets of the channel.
Step 3: At T^{th} iteration
Step 3.1: Find a list *List_A* of non-overlapping nets having highest total *parametric difference* from *Net_List* without violation of vertical constraints.
Topmost_track ← *Topmost_track* + 1
If all the nets of *List_A* are initially assigned to *Topmost_track* of the routing solution of *MTAH* then
Reassign these nets to *Topmost_track* of new routing solution.
Else
Assign the nets of *List_A* to *Topmost_track* of the new routing solution.
Step 3.2: If *Topmost_track* is fully utilized by nets, then go to Step 3.3.
Else
Find another list, *List_B* of non-overlapping intervals from *Net_List* (other than nets of *List_A*) which may be assigned along with the nets of *List_A* and does not reduce total *parametric difference* of *Topmost_track*.
Assign the nets of *List_B* to *Topmost_track* of the new routing solution.
Step 3.3: Delete all nets of *List_A* and *List_B* from *Net_List* (or freeze these nets in *Net_List* for reconsideration).
Step 3.4: *List_A* ← NULL
List_B ← NULL
Delete the corresponding vertices and all its connecting edges of the corresponding nets from VCG and HNCG. Go for next iteration.
Step 4: Total number of required tracks is *Topmost_track*.
Find the total vertical wire length of the new routing solution.
Step 5: EXIT

Fig. 1. Algorithm *Further_Reduced_Wire_Length (FRWL)*.

Algorithm *Multi-Layer_Reduced_Wire_Length (MLRWL)*

Input: A two-layer routing solution of a channel specification using algorithm *FRWL* for wire length minimization and an integer *i* (as the number of horizontal layers of multi-layer routing model V_iH_i, $2 \le i < d_{max}$).

Output: Multi-layer reduced wire length routing solution in V_iH_i, $2 \le i < d_{max}$ routing model.

Step 1: *T* denotes the total number of tracks in the two-layer routing solution using algorithm *FRWL*.
Step 2: *K* ← 0
I ← 1
Step 3: While $I \le T$ do
K ← *K* + 1
J ← 1
While $J \le i$ and $I \le T$ do
Assign the nets of I^{th} track of the two-layer routing solution using *FRWL* to K^{th} track of the *J*-th horizontal layer in the V_iH_i, $2 \le i < d_{max}$ multi-layer routing model.
J ← *J* + 1
I ← *I* + 1
Step 4: Apply algorithm *FRWL* separately on *i* pairs of VH layers in order to compute multi-layer V_iH_i, $2 \le i < d_{max}$ routing solution with further reduced wire length.
Step 5: Check whether there is any further possibility to reduce total vertical wire length of nets assigned to the first $i - 1$ horizontal layers (as these layers are flanked by vertical layers).
Step 6: Eliminate the common unused tracks of different horizontal layers, if any, and compute the final multi-layer (V_iH_i, $2 \le i < d_{max}$) routing solution with minimizing wire length.
Step 7: Exit

Fig. 2. Algorithm *Multi-Layer_Reduced_Wire_Length (MLRWL)*.

As an example run, we consider the most important channel instance *Deutsch's Difficult Example* (*DDE*) which is used several times in literature to test new algorithms in channel routing problem [9]. *DDE* requires 29 tracks and 3694 units of total vertical wire length using algorithm *FRWL* to compute further reduced wire length two-layer (VH) routing solution and for the same routing model *DDE* requires 29 tracks and 3845 units of total vertical wire length using algorithm *TAH* for area minimization; it requires 29 tracks and 3820 units of total vertical wire length using algorithm *TAH* for wire length minimization; it requires 31 tracks and 3912 units of total vertical wire length using algorithm for wire length minimization developed by Mitra *et* al.; and it requires 30 tracks and 3959 units of total vertical wire length using algorithm for wire length minimization developed by Saha Sau *et* al.

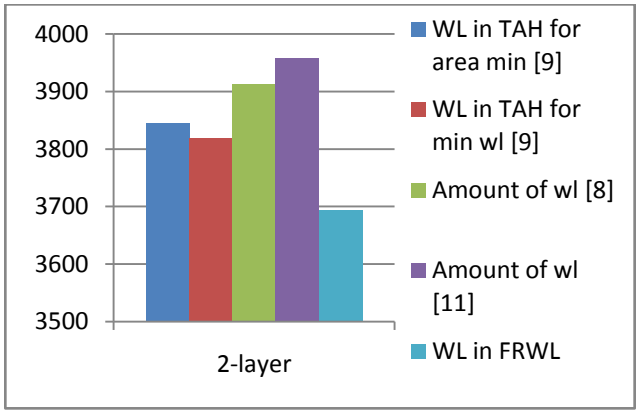

Fig. 3. A bar chart representing the performance of two-layer further reduced wire length routing solution for channel instance *DDE* using algorithm *FRWL* compare to *TAH* for area and *TAH* for wire length min, Algorithm developed by Mitra *et al*, and Algorithm developed by Saha Sau *et al*.

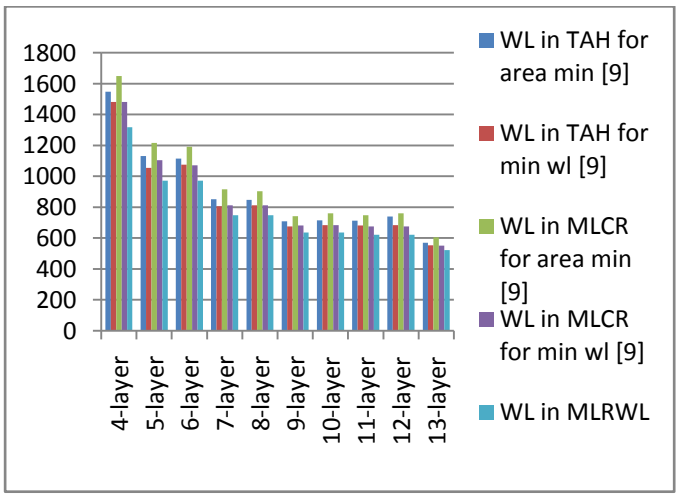

Fig. 4. A bar chart representing the performance of multi-layer routing solution using the algorithm *MLRWL* for channel instance *DDE*

compare to *TAH* for area and *TAH* for wire length min, *MLCR* for area minimization, and *MLCR* for wire minimization.

Here we have shown the performance of multi-layer (V_iH_i, $2 \leq i < d_{max}$ and V_iH_{i+1}, $2 \leq i < d_{max} - 1$) routing algorithm *MLRWL* compare to the algorithm *TAH* for area minimization, the algorithm *TAH* for wire length minimization, the algorithm *MLCR* for area minimization, and the algorithm *MLCR* for wire length minimization computed in same model considering two channel instances *DDE* and *r2* in Fig. 4. and Fig. 5., respectively [9].

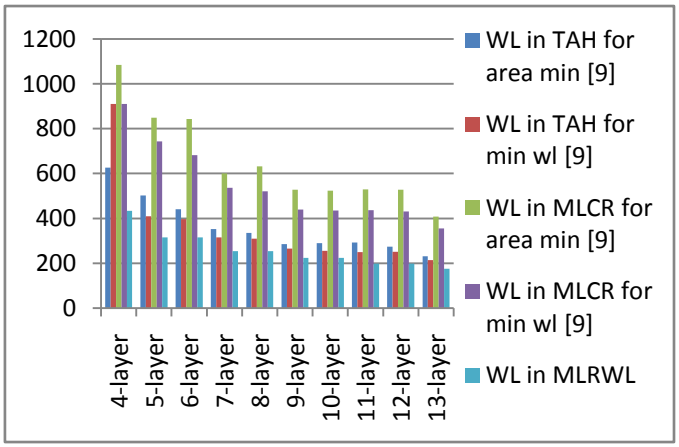

Fig. 5. A bar chart representing the performance of multi-layer routing solution using the algorithm *MLRWL* for channel instance *r2* compare to *TAH* for area and *TAH* for wire length min, *MLCR* for area minimization, and *MLCR* for wire minimization.

IV. Experimental Results

In this section, we show some experimental results that are computed using our algorithms *FRWL* for two-layer no-dogleg routing and *MLRWL* for four- through thirteen-layer no-dogleg routing developed in Section 3 of this paper. A very few experimental results is available in literature in the assumed routing model. The results of the newly designed algorithm for existing benchmark instances [9,14] are included in Table 1.

In Table 1, we compare our experimental results (of computing total vertical wire length) after implementation of algorithm *FRWL* with the results computed using algorithm *Track_Assignment_Heuristic* (*TAH*) for area minimization [9], *TAH* for wire length minimization [9], algorithm developed by Mitra *et al*. for wire length minimization [8], and algorithm *Modified_Track_Assignment_Heuristic* (*MTAH*) for wire length minimization [11]. All these algorithms have been developed for computing area and/or wire length minimization in two-layer no-dogleg Manhattan channel routing model [8,9,11].

We have compared our experimental results (of minimizing total vertical wire length) after implementation of algorithm *MLRWL* developed in four-layer through thirteen-layer no-dogleg Manhattan routing model with the results computed using algorithm *TAH* for area minimization [9], *TAH* for wire length minimization [9], algorithm *Multi-Layer_Channel_Router* (*MLCR*) for area minimization [9], and *MLCR* for wire length minimization [9] in respective layer of routing model. For all the cases *MLRWL* performs very well. Here we have shown some of those results. In Table 2, we compare our experimental results (of minimizing total vertical wire length) after implementation of algorithm *MLRWL* developed in four-layer no-dogleg Manhattan routing model with the results computed using algorithm *TAH* in four-layer routing model for area minimization [9], *TAH* in four-layer routing model for wire length minimization [9], algorithm developed by Fang *et al*. in four-layer routing model for wire length minimization [2] and algorithm *Multi-Layer_Channel_Router* (*MLCR*) in four-layer routing model for area minimization [9], and *MLCR* in four-layer routing model for wire length minimization [9].

In Tables 3 through 7 we also compare our associated experimental results (with total vertical wire length) computed using algorithm *MLRWL* developed in six-, seven-, ten-, twelve-, and thirteen-layer no-dogleg Manhattan routing models, respectively, with the results computed using algorithm *TAH* for area minimization [9], *TAH* for wire length minimization [9], *MLCR* for area minimization [9], and *MLCR* for wire length minimization [9] that have also been developed in the respective layers of interconnect, respectively.

Though the channel instances considered in this paper are a bit older, but incidentally these are the most well-known benchmark channel instances for last couple of decades for computing experimental results computed by the researchers of this domain of work. For the same reason, we consider these instances to judge the results computed using algorithms *FRWL* and *MLRWL* developed herein with the existing results available in literature for several existing algorithms as mentioned above.

It is interesting to note that the routing solutions computed using the algorithms developed in this paper mostly reduce total (vertical) wire length in the routing solutions computed using them, though some of them have taken one or two more tracks than the number of tracks required for the same channels in the existing literature. It is not possible to include the recent papers as in the said routing model; these are the only existing results for the assumed problem under consideration.

We have implemented the algorithms in a computing environment of Dev-C++ 4.9.9.1 on Intel(R) core(TM)2 Duo T6400 with 2.00 GHz clock. We also observe that for all the channel instances, the required CPU time for computing all these results is negligibly small, that varies from 15.01×10^{-9} seconds and 40.34×10^{-9} seconds. Time for each instance is calculated by taking the difference between the start and the end time of program execution of that instance.

V. CONCLUSION

In this paper we have developed two algorithms to compute the routing solutions in the two- and multi-layer reserved no-dogleg Manhattan channel routing model. The two- and multi-layer channel routing problem of minimizing wire length is NP-hard [9,14]. Here we have developed polynomial time computable heuristic algorithms that reduce total wire length channel routing solutions in two and multi-layer channel routing models. Our devised algorithms *FRWL* and *MLRWL* effectively compute mostly reduced (wire length) feasible routing solutions in the said models. The time complexity of both the algorithms developed herein is $O(t(n+e))$, where t, n, and e denote the number of tracks to route the channel, the number of nets in the given channel instance, and the size of HNCG, respectively. The performance of our algorithms is very much promising. Now we may view the problem of reducing wire length along with reducing area for computing dogleg and no-dogleg routing solutions in the two- and multi-layer channel routing models.

REFERENCES

[1] C. J. Alpert, D. P. Mehta, S. S. Sapatnekar, *Handbook of Algorithms for Physical Design Automation*, CRC Press, London, New York, 2009.

[2] S.-C. Fang, W.-S. Feng, S.-L. Lee, A new Efficient Approach to Multilayer Channel Routing Problem, *Proc. of 29th ACM/IEEE Design Automation Conference*, pp. 579-584, 1992 .

[3] M. D. Formann Wagner, F. Wagner, Routing Through a Dense Channel with Minimum Total Wire Length, *Proc. of Second Annual ACM-SIAM Symposium*, pp. 475-482, 1991.

[4] M. C. Golumbic, Algorithmic Graph Theory and Perfect Graphs, Academic Press, New York, 1980.

[5] A. Hashimoto, J. Stevens, Wire Routing by Optimizing Channel Assignment within Large Apertures, *Proc. of Eighth ACM Design Automation Workshop*, pp. 155-169, 1971.

[6] C. Hong, Y. Kim, The Efficient Hybrid Approach to Channel Routing Problem, *Intl. Journal of Advanced Science and Technology*, vol. 42, pp. 61-68, 2012.

[7] J. Lienig, Introduction to Electromigration-aware Physical Design (Invited talk), *Proc. of ISPD'06*, pp. 39-46, 2006.

[8] P. Mitra, N. Ghoshal, R. K. Pal, A Graph Theoretic Approach to Minimize Total Wire Length in Channel Routing, *Proc. of 18th IEEE Region 10 Intl. Conference on Convergent Technologies for the Asia-Pacific (IEEE TENCON 2003)*, vol. 1, pp. 414-418, 2003.

[9] R. K. Pal, *Multi-Layer Channel Routing: Complexity and Algorithms*, Narosa Publishing House, New Delhi (Also published from CRC Press, Boca Raton, USA and Alpha Science International Ltd., UK), 2000.

[10] R. K. Pal, A. K. Datta, S. P. Pal, M. M. Das, A. Pal, A General Graph Theoretic Framework for Multi-Layer Channel Routing, *Proc. of Eighth VSI/IEEE Intl. Conference on VLSI Design*, New Delhi, India, pp. 202-207, 1995.

[11] S. Saha Sau, R. Pal, An Efficient High Performance Parallel Algorithm to Yield Reduced Wire Length VLSI Circuits, *Proc. of Fifth Intl. Conference on Computers and Devices for Communication (CODEC 2012)*, 2012.

[12] S. Saha Sau, A. Pal, T. N. Mandal, A. K. Datta, R. K. Pal, A. Chaudhuri, A Graph based Algorithm to Minimize Total Wire Length in VLSI Channel Routing, *Proc. of 2011 IEEE Intl. Conference on Computer Science and Automation Engineering (CSAE)*, vol. 3, pp. 61-65, 2011.

[13] K. A. Somogyi, A. Recski, On the Complexity of the Channel Routing Problem in the Dogleg-free Multi-layer Manhattan Model, *ACTA Polytechnica Hungarica*, vol. 1, no. 2, 2004.

[14] T. Yoshimura, E. S. Kuh, Efficient Algorithms for Channel Routing, *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 1, pp. 25-35, 1982.

TABLE 1
PERFORMANCE OF ALGORITHM *FRWL* FOR COMPUTING TWO-LAYER NO-DOGLEG REDUCED WIRE LENGTH ROUTING SOLUTIONS IN COMPARISON TO OTHER ROUTING SOLUTIONS COMPUTED IN TWO-LAYER NO-DOGLEG CHANNEL ROUTING [8,9,11]

Channel Instance	d_{max}	v_{max}	# tracks using *TAH* [9]	WL in *TAH* for area min [9]	WL in *TAH* for min wl [9]	# tracks for wl min [8]	Amount of wl [8]	# tracks for wl min [11]	Amount of wl [11]	# tracks using *FRWL*	Amount of wl using *FRWL*	% red. of *FRWL* over *TAH* for area min [9]	% red. of *FRWL* over *TAH* for wl min [9]	% red. of *FRWL* over algo in [8]	% red. of *FRWL* over algo in [11]
Ex. 1	12	7	12	236	235	13	236	12	231	12	225	4.66	4.26	4.66	2.60
Ex. 2	15	4	15	482	456	16	404	16	436	15	371	23.03	18.64	8.17	14.91
Ex. 3(a)	15	4	16	694	630	17	528	17	541	17	505	27.23	19.84	4.36	6.65
Ex. 3(b)	17	9	18	853	806	19	819	20	852	18	761	10.79	5.58	7.08	10.68
Ex. 3(c)	18	6	19	1106	1026	18	863	20	976	18	832	24.77	18.91	3.59	14.75
Ex. 4(b)	17	13	19	1483	1470	20	1305	19	1458	19	1251	15.64	15.64	4.14	14.20
Ex. 5	20	3	20	1699	1598	21	1130	20	1151	20	1071	36.96	32.98	5.22	6.95
DDE	19	23	29	3845	3820	31	3912	30	3959	29	3694	3.93	3.30	5.57	6.69
r1	20	6	23	1708	1625	24	1356	26	1580	24	1349	21.02	16.99	0.52	14.62
r2	20	5	20	1081	989	22	781	22	781	22	765	29.23	22.65	2.05	2.05
r3	16	7	18	1345	1315	19	1104	20	1194	19	1066	20.74	18.94	3.44	10.72
r4	15	7	18	1598	1546	19	1448	19	1626	19	1421	11.08	8.09	1.87	12.61
Ex. 3(b).1	17	12	21	1059	1024	21	895	20	904	20	849	19.83	17.09	5.14	6.08
Ex. 3(c).1	18	6	18	1040	972	18	900	20	976	18	832	20.00	14.40	7.56	14.75

TABLE 2
PERFORMANCE OF ALGORITHM *MLRWL* FOR COMPUTING FOUR-LAYER NO-DOGLEG REDUCED WIRE LENGTH ROUTING SOLUTIONS IN COMPARISON TO OTHER ROUTING SOLUTIONS COMPUTED IN FOUR-LAYER NO-DOGLEG CHANNEL ROUTING [2,9,10]

Channel Instance	# tracks using *TAH* [9]	WL in *TAH* for area min [9,10]	WL in *TAH* for min wl [9,10]	# tracks for wl min [2]	Amount of wl [2]	# tracks using *MLCR* [9]	WL in *MLCR* for area min [9]	WL in *MLCR* for min wl [9]	# tracks using *MLRWL*	Amount of wl using *MLRWL*	% red. of *MLRWL* over *TAH* for area min [9,10]	% red. of *MLRWL* over *TAH* for min wl [9,10]	% red. of *MLRWL* over algo in [2]	% red. of *MLRWL* over *MLCR* for area min [9]	% red. of *MLRWL* over *MLCR* for min wl [9]
Ex. 1	6	175	148	6	180	6	183	148	6	119	32.00	19.59	33.89	34.97	19.60
Ex. 2	8	267	236	-	-	8	325	236	8	212	20.60	10.17	-	34.77	10.17
Ex. 3(a)	8	404	335	-	-	8	433	335	9	283	29.95	15.52	-	34.64	15.52
Ex. 3(b)	9	535	469	9	541	9	531	469	9	407	23.93	13.22	24.77	23.35	13.22
Ex. 3(c)	9	613	521	9	594	9	638	521	9	447	27.08	14.20	24.75	29.94	14.20
Ex. 4(b)	9	923	842	9	856	9	977	842	10	660	28.49	21.62	22.90	32.45	21.62
Ex. 5	10	861	738	10	800	10	885	738	10	580	32.64	21.41	27.50	34.46	21.41
DDE	10	1547	1481	10	1559	10	1648	1481	10	1318	14.80	11.01	15.46	20.02	11.01
r1	11	1021	1055	-	-	10	1195	1055	12	717	29.78	32.04	-	40.00	32.04
r2	10	627	910	-	-	10	1084	910	12	434	30.78	52.31	-	59.96	52.31
r3	9	773	910	-	-	8	971	910	10	585	24.32	35.71	-	39.75	35.71
r4	8	1015	969	-	-	8	1130	969	10	783	22.86	19.20	-	30.71	19.20
Ex. 3(b).1	9	559	462	-	-	9	558	462	10	448	19.86	3.03	-	19.71	3.03
Ex. 3(c).1	9	613	678	-	-	9	777	678	9	447	27.08	34.07	-	42.47	34.07

TABLE 3
PERFORMANCE OF ALGORITHM *MLRWL* FOR COMPUTING SIX-LAYER NO-DOGLEG REDUCED WIRE LENGTH ROUTING SOLUTIONS IN COMPARISON TO OTHER ROUTING SOLUTIONS COMPUTED IN SIX-LAYER NO-DOGLEG CHANNEL ROUTING [9,10]

Channel Instance	# tracks using *TAH* [9]	WL in *TAH* for area min [9,10]	WL in *TAH* for min wl [9,10]	# tracks using *MLCR* [9]	WL in *MLCR* for area min [9]	WL in *MLCR* for min wl [9]	# tracks using *MLRWL*	Amount of wl using *MLRWL*	% red. of *MLRWL* over *TAH* for area min [9,10]	% red. of *MLRWL* over *TAH* for min wl [9,10]	% red. of *MLRWL* over *MLCR* for area min [9]	% red. of *MLRWL* over *MLCR* for min wl [9]
Ex. 1	4	146	113	4	127	112	4	95	34.93	15.93	25.20	15.18
Ex. 2	5	190	159	5	210	163	5	147	22.63	7.55	30.00	9.82
Ex. 3(a)	5	252	226	5	277	214	6	209	17.06	7.52	24.55	2.34
Ex. 3(b)	6	372	335	6	386	325	6	293	21.24	12.54	24.09	9.85
Ex. 3(c)	6	451	366	6	479	372	6	324	28.16	11.48	32.36	12.90
Ex. 4(b)	6	685	597	6	730	602	7	483	29.49	19.10	33.84	19.77
Ex. 5	7	628	544	7	718	549	7	427	32.01	21.51	40.53	22.22
DDE	7	1114	1075	7	1191	1071	7	971	12.84	9.67	18.47	9.34
r1	7	683	595	7	916	768	8	515	24.60	13.45	43.78	32.94
r2	7	441	397	7	843	682	8	315	28.57	20.66	62.63	53.81
r3	6	563	496	6	759	701	7	438	22.20	11.69	42.29	37.52

r4	5	676	565	5	760	664	7	585	13.46	-3.42	23.03	11.90
Ex. 3(b).1	6	370	328	6	398	329	7	324	12.43	1.22	18.59	1.52
Ex. 3(c).1	6	451	366	6	554	476	6	324	28.16	11.48	41.52	31.93

TABLE 4
PERFORMANCE OF ALGORITHM *MLRWL* FOR COMPUTING SEVEN-LAYER NO-DOGLEG REDUCED WIRE LENGTH ROUTING SOLUTIONS IN COMPARISON TO OTHER ROUTING SOLUTIONS COMPUTED IN SEVEN-LAYER NO-DOGLEG CHANNEL ROUTING [9,10]

Channel Instance	# tracks using *TAH* [9]	WL in *TAH* for area min [9,10]	WL in *TAH* for min wl [9,10]	# tracks using *MLCR* [9]	WL in *MLCR* for area min [9]	WL in *MLCR* for min wl [9]	# tracks using *MLRWL*	Amount of wl using *MLRWL*	% red. of *MLRWL* over *TAH* for area min [9,10]	% red. of *MLRWL* over *TAH* for min wl [9,10]	% red. of *MLRWL* over *MLCR* for area min [9]	% red. of *MLRWL* over *MLCR* for min wl [9]
Ex. 1	3	101	89	3	126	89	3	75	25.74	15.73	40.48	15.73
Ex. 2	3	132	114	3	142	113	3	105	20.46	7.90	26.06	7.08
Ex. 3(a)	3	168	154	3	190	151	4	153	8.93	0.65	19.47	-1.33
Ex. 3(b)	4	267	241	4	279	235	4	213	20.23	11.62	23.66	9.36
Ex. 3(c)	4	316	269	4	324	267	4	238	24.68	11.52	26.54	10.86
Ex. 4(b)	4	493	432	4	510	436	4	323	34.48	25.23	36.67	25.92
Ex. 5	4	414	346	4	454	355	4	289	30.19	16.47	36.34	18.59
DDE	4	715	684	4	760	683	4	636	11.05	7.02	16.32	6.88
r1	4	453	369	4	576	492	5	357	21.19	3.25	38.02	27.44
r2	4	289	255	4	524	435	5	224	22.49	12.16	57.25	48.51
r3	4	392	363	4	545	508	4	294	25.00	19.01	46.06	42.13
r4	3	436	384	3	497	440	4	391	10.32	-1.82	21.33	11.14
Ex. 3(b).1	4	259	238	4	285	236	4	209	19.31	12.19	26.67	11.44
Ex. 3(c).1	4	316	269	4	378	338	4	238	24.68	11.52	37.04	29.59

TABLE 5
PERFORMANCE OF ALGORITHM *MLRWL* FOR COMPUTING TEN-LAYER NO-DOGLEG REDUCED WIRE LENGTH ROUTING SOLUTIONS IN COMPARISON TO OTHER ROUTING SOLUTIONS COMPUTED IN TEN-LAYER NO-DOGLEG CHANNEL ROUTING [9,10]

Channel Instance	# tracks using *TAH* [9]	WL in *TAH* for area min [9,10]	WL in *TAH* for min wl [9,10]	# tracks using *MLCR* [9]	WL in *MLCR* for area min [9]	WL in *MLCR* for min wl [9]	# tracks using *MLRWL*	Amount of wl using *MLRWL*	% red. of *MLRWL* over *TAH* for area min [9,10]	% red. of *MLRWL* over *TAH* for min wl [9,10]	% red. of *MLRWL* over *MLCR* for area min [9]	% red. of *MLRWL* over *MLCR* for min wl [9]
Ex. 1	2	81	70	2	83	75	2	66	18.52	5.71	20.48	12.00
Ex. 2	3	133	113	3	138	112	3	102	23.31	9.74	26.08	8.93
Ex. 3(a)	3	163	149	3	189	148	3	133	18.41	10.74	29.63	10.14
Ex. 3(b)	3	219	197	3	225	192	3	178	18.72	9.65	20.89	7.29
Ex. 3(c)	3	254	220	3	253	221	3	200	21.6	9.09	20.95	9.50
Ex. 4(b)	3	385	354	3	378	351	3	279	27.53	21.19	26.19	20.51
Ex. 5	4	422	348	4	454	349	4	282	33.18	18.97	37.89	19.20
DDE	4	739	684	4	760	675	4	621	15.97	9.21	18.29	8.00
r1	4	460	402	4	570	484	4	308	33.04	23.38	45.97	36.36
r2	4	274	251	4	528	431	4	199	27.37	52.59	54.92	72.39
r3	3	320	292	3	434	412	3	251	21.56	14.04	42.17	39.08
r4	3	437	381	3	509	435	3	333	23.80	12.60	34.58	23.45
Ex. 3(b).1	3	211	194	3	230	194	3	177	16.11	8.76	23.04	8.76
Ex. 3(c).1	3	254	220	3	306	276	3	200	21.6	9.09	34.64	27.54

TABLE 6
PERFORMANCE OF ALGORITHM *MLRWL* FOR COMPUTING TWELVE-LAYER NO-DOGLEG REDUCED WIRE LENGTH ROUTING SOLUTIONS IN COMPARISON TO OTHER ROUTING SOLUTIONS COMPUTED IN TWELVE-LAYER NO-DOGLEG CHANNEL ROUTING [9,10]

Channel Instance	# tracks using *TAH* [9]	WL in *TAH* for area min [9,10]	WL in *TAH* for min wl [9,10]	# tracks using *MLCR* [9]	WL in *MLCR* for area min [9]	WL in *MLCR* for min wl [9]	# tracks using *MLRWL*	Amount of wl using *MLRWL*	% red. of *MLRWL* over *TAH* for area min [9,10]	% red. of *MLRWL* over *TAH* for min wl [9,10]	% red. of *MLRWL* over *MLCR* for area min [9]	% red. of *MLRWL* over *MLCR* for min wl [9]
Ex. 1	2	79	70	2	81	77	2	66	16.46	5.71	18.52	14.29
Ex. 2	3	133	113	3	133	111	3	102	23.31	9.73	23.31	8.11
Ex. 3(a)	3	163	149	3	183	149	3	130	20.25	12.75	28.96	12.75
Ex. 3(b)	3	220	195	3	221	192	3	175	20.45	10.26	20.81	8.85
Ex. 3(c)	3	256	219	3	261	219	3	198	22.66	9.59	24.14	9.59
Ex. 4(b)	3	382	351	3	408	356	3	274	28.27	21.94	32.84	23.03
Ex. 5	3	319	286	4	368	290	3	241	24.45	15.73	34.51	16.90
DDE	3	570	553	4	605	550	3	522	8.42	5.82	13.72	5.09
r1	3	356	317	4	459	399	4	303	14.89	4.42	33.99	24.06
r2	3	231	214	4	408	356	3	175	24.24	18.22	57.11	50.84
r3	3	320	292	3	433	411	3	248	22.50	15.07	42.73	39.66
r4	3	437	381	3	485	440	3	325	25.63	14.70	32.99	26.14
Ex. 3(b).1	3	210	194	3	219	194	3	174	17.14	10.31	20.09	10.31
Ex. 3(c).1	3	256	219	3	302	273	3	198	22.66	9.59	34.44	27.47

TABLE 7
PERFORMANCE OF ALGORITHM *MLRWL* FOR COMPUTING THIRTEEN-LAYER NO-DOGLEG REDUCED WIRE LENGTH ROUTING SOLUTIONS IN COMPARISON TO OTHER ROUTING SOLUTIONS COMPUTED IN THIRTEEN-LAYER NO-DOGLEG CHANNEL ROUTING [9,10]

Channel Instance	# tracks using *TAH* [9]	WL in *TAH* for area min [9,10]	WL in *TAH* for min wl [9,10]	# tracks using *MLCR* [9]	WL in *MLCR* for area min [9]	WL in *MLCR* for min wl [9]	# tracks using *MLRWL*	Amount of wl using *MLRWL*	% red. of *MLRWL* over *TAH* for area min [9,10]	% red. of *MLRWL* over *TAH* for min wl [9,10]	% red. of *MLRWL* over *MLCR* for area min [9]	% red. of *MLRWL* over *MLCR* for min wl [9]
Ex. 1	3	107	90	3	119	93	3	77	28.04	14.44	35.29	17.20
Ex. 2	4	154	142	4	173	137	4	125	18.83	11.97	27.75	8.76
Ex. 3(a)	4	228	189	4	239	182	4	162	28.95	14.29	32.22	10.99
Ex. 3(b)	5	319	279	5	326	284	5	250	21.63	10.39	23.31	11.97
Ex. 3(c)	5	385	316	5	376	323	5	279	27.53	11.71	25.80	13.62
Ex. 4(b)	5	597	513	5	597	520	5	380	36.35	25.93	36.35	26.92
Ex. 5	5	478	415	5	478	423	5	337	29.50	18.80	29.50	20.33
DDE	5	852	806	5	915	811	5	748	12.21	7.20	18.25	7.77
r1	6	597	511	5	679	591	6	409	31.49	19.96	39.76	30.80
r2	5	352	315	5	601	537	6	254	27.84	19.37	57.74	52.70
r3	5	425	383	4	566	518	5	344	19.06	10.18	39.22	33.59
r4	4	564	480	4	622	542	5	455	19.33	5.21	26.85	16.05
Ex. 3(b).1	5	331	291	5	338	288	5	249	24.77	14.43	26.33	13.54
Ex. 3(c).1	5	385	317	5	467	405	5	279	27.53	11.99	40.26	31.11