An Evolutionary 3D Over−the−Cell Router

B. M. Goni and T. Arslan

ABSTRACT

This paper reports on a new algorithm which carries out combined 3D (4−layer) and over−the−cell (OTC) routing in order to reduce chip area in 3D VLSI technology. In order to cope with the complexity of the search space, the algorithm is tailored around a class of heuristic techniques called genetic algorithms. Our results indicate up to 62% savings in chip area when compared to conventional multilayer channel routers, using several internationally well known benchmarks.

1. INTRODUCTION

With silicon VLSI technology approaching its fundamental scaling limit of 0.2 µm, the concept of three−dimensional (3D) integration has been proposed to enhance packing density and speed performance [1]. Also performance and cost of the widely used submicron 2D VLSI technology are primarily determined by interconnection delays and on−chip area. 3D VLSI offers the possibility of overcoming this problem [1]. Hence being the ideal technology for complex system−on−chip (SoC) applications. A 3D VLSI chip consists of a stack of active (silicon) layers made possible by the silicon on insulator (SOI) technology. In this technology, devices are first fabricated on an active layer using a 2D process. A passive (silicon dioxide) layer is then grown on top of the active layer and planarized by etching before another polysilicon layer is deposited on the top of the insulator and recrystallized by laser beam. The new silicon layer can then be used to fabricate devices using a 2D process. By repeating these steps, many active layers can be packed into one chip.

The use of 3D VLSI technology allows more layers for routing. Thus, the need for developing multilayer routing algorithms. In [2], a multi−layer channel router called *chameleon* has been developed. Chameleon is based on YACR2. The main feature of Chameleon is that it uses a general approach for multilayer channel routing. Stacked vias can be included or excluded, and separate design rules for each layer can be specified. The Chameleon consists of two stages; a partitioner and a detailed router. The partitioner divides the problem into two and three−layer subproblems such that the global channel area is minimized. The detailed router then implements the connections using generalizations of the algorithms employed in YACR2.

In [3] a new approach to the three− or four−layer channel routing problem is presented. A general technique is developed which transforms a two layer solution systematically into a three−layer routing solution. This router is shown to perform better than other previously published three−layer channel routers.

The idea of using the areas over the cells that are adjacent to channels has been put into practice in order to further reduce the total area of channels [4]. Such areas are called over−the−cell (OTC) routing areas. Like a channel, an OTC routing area consists of tracks and columns. The aim of OTC channel routing is to minimize the congestion (number of tracks used) inside the channels. In light of the advancement of multilayer processes more OTC area is now available for routing and hence, further reduction in layout can be accomplished. [5] presented an efficient four layer over−the−cell router, for a cell model similar to Target Based Cell (TBC), called Arbitrary Terminal Model(ATM). In this cell model terminals can be placed at any arbitrary locations in the cell.

A number of techniques have been proposed in the literature for solving the channel routing problem [4,7]. However, as circuit complexities increase, new techniques are necessary to tackle the complexity issue. Genetic algorithms (GAs) which are a new class of heuristic search methods based on biological evolution in nature have been applied increasingly successfully to find good heuristic solutions to NP−complete optimization problems [8, 9].

This paper reports on a new algorithm which carries out combined 3D (4−layer) and over−the−cell (OTC) routing in order to reduce chip area in 3D VLSI technology. In order to cope with the complexity of the search space in 3D VLSI technology, the algorithm is tailored around a class of heuristic techniques called genetic algorithms. The algorithm operates in two stages. In the first a two layer netlist is obtained using 2−layer OTC routing. This in turn is fed into the next stage of the algorithm, which partitions the netlist into two sets (top and bottom) with the aim of minimizing area. 4−layer OTC routing starts by identifying all multi−terminal net circuit segments (in top and bottom partitions) for over−the−cell routing, and places these on tracks over−the−cell according to horizontal constraints. The algorithm proceeds to route the remaining net circuits inside the main channel areas of the top and bottom partitions subject to horizontal and vertical constraints. This procedure ensures that only two terminal nets are left within the channel after its action. This reduces the amount of congestion in the channel area. Our results indicate up to 62% savings in chip area when compared to conventional multilayer channel routers, using several internationally well known benchmarks. In addition the 4−layer results of our algorithm is up to 50% better than the 5−layer results achieved by the best of previous multilayer routers.

2. THE ROUTING PROBLEM DEFINITION

A channel routing problem is traditionally presented by specifying a netlist such as the Yoshimura−Kuh (YK) channel netlist in [7]. The realization of this netlist for two layers using our OTC router is shown in Figure 1, with two rows of terminals along its top and bottom sides. Each terminal is assigned a number from the interval [0, N]. Terminals with the same positive number *i* have to be connected and form *net i*. Unconnected terminals are designated by zero. Two wiring layers are available for each partition (2−layer sub−problem), one for horizontal segments and the other for vertical segments. The horizontal segment of each net must be placed on a horizontal track [7]. Wire transitions between one layer to the next are made through vias.

The channel router is required to assign each net to a horizontal track subject to vertical constraints and horizontal constraints, in addition to minimizing such horizontal tracks [7]. Also, the horizontal segment of each net must be placed on a single horizontal track and may not be segmented over several tracks.

In the over−the−cell channel areas, there are only horizontal constraints and vertical constraints do not exist. We assume that vias are allowed in the over−the−cell areas.

A vertical constraint or violation [7] is said to exist between two distinct nets if the terminal number of the top connected net is the same with that of the bottom connected net. In such a case the former net is placed above the latter net. Horizontal constraints or violations [7] occur between two different nets if the sets of terminal numbers spanned by the two nets intersect. For example if we consider the sets of terminal numbers spanned by nets 2 and 5 in Figure 1, we see that the span of net 5 (3−5) is contained in 2. The two nets intersect and therefore cannot be routed on the same horizontal track.

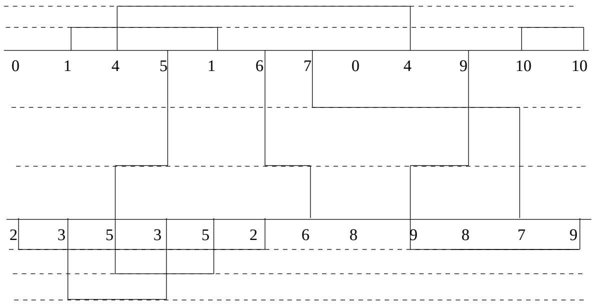

Figure 1: YK [7] channel solution obtained with our 2−layer OTC router

3. GENETIC ALGORITHMS

Genetic algorithms are search techniques which are based on the mechanics of natural selection and the principle of *survival of the fittest* [8, 10]. They operate on a population of structures which are fixed length strings representing possible solutions of a given optimization problem. A population of solutions is maintained and successive *generations* are produced by manipulating the solutions in the current population. Each solution has a *fitness* that measures its *competence*. New solutions are formed typically by merging two previous ones via a *crossover* operator. Other new solutions are simply modifications of previous ones, using a *mutation* operator. Successive generations are produced with new solutions probabilistically replacing older ones based on relative fitness. An ad hoc termination condition is often used and the best solution is usually reported. GAs have several advantages over traditional search and optimization algorithms. These advantages stem in part from its ability to maintain simultaneously information about a variety of points in the solution space. This helps prevent the GA from being trapped at inferior local minima. Another feature of GAs is their use of building blocks in creating new solutions. This allows a GA to take advantage of the high quality sub−solutions that may already be present in existing solutions.

4. THE OPTIMIZATION ALGORITHM

Our algorithm is implemented within the framework of GALOPPS ("Genetic Algorithm Optimized for Portability and Parallelism" System) [9], which is a generic tool for the implementation of genetic systems. The developed algorithm is based on a model developed by Goldberg [8]. Our investigations revealed that with a population of 100 chromosomes, the following GA characteristics are necessary for a performance which provides a \`good´ trade off in speed and quality of solutions:

1. Uniform crossover and random multibit mutation at rates of 0.6 and 0.05.
2. Chromosome selection procedure using Baker´s Stochastic Universal Sampling algorithm [8].
3. Employment of an elitist selection strategy [9].

The chromosome structure for a given routing problem is presented as follows;

[TOP, INT, BOT, ID]

where, for each net,
TOP = top over−the−cell track number.
INT = internal track number.
BOT = bottom over−the−cell track number.
ID = net number.

A key component of the GA is the fitness evaluation stage, where the quality of the different routing solutions are assessed. After initialization the fitness function (see Figure 2) starts by partitioning a 2−layer solution netlist (see Figure 1) into two sets top and bottom. Each of these partitions corresponds to a two layer sub−problem. The nets on these partitions are then routed separately. The fitness function then proceeds to identify all multi−terminal net circuits (on each partition) for over−the−cell routing, and places these on tracks over−the−cell according to horizontal constraints. Further, it proceeds to route the remaining net circuits inside the channel subject to horizontal and vertical constraints. For a given structure Y, the fitness function F(Y) can be specified as:

$$F(Y) = OTCP(Y) + HVP(Y) + VVP(Y).$$

Where OTCP, HVP and VVP are routing procedures and are described as follows:

4.1 Over−the−cell routing (OTCP): Performs the routing of multi−terminal net circuits in the over−the−cell areas (top and bottom partitions), and returns their number. After this procedure only net circuits with single pin connections at both the top and bottom will remain inside the main channel areas (top and bottom partitions).

4.2 Horizontal violation elimination (HVP): Carries out the elimination of horizontal violations, by returning the number of net circuits without such violations. It is applied to both the over−the−cell and main channel areas for the two partitions top and bottom.

4.3 Vertical violation elimination (VVP): Performs the elimination of vertical violations, by returning the number of two pin (terminal) net circuits connected to the top and bottom pins inside the main channel areas (top and bottom partitions) without such violations. As there are no vertical violations in the regions over−the−cell, this procedure is not applied.

Two or more net circuits are placed on the same track if they are without horizontal and vertical violations in the main channel area. For the over−the−cell areas only the elimination of horizontal violations need to be satisfied before net circuits are routed on the same track. This is illustrated using the Yoshimura−Kuh channel circuit in [7], the algorithm starts by first identifying top and bottom multi−terminal net circuits in all 2−layer partitions and routes them over−the−cell (nets 1, 2, 3, 4, 5, 8, 9 and 10) as shown in Figure 3(a) and (b). It then carries out routing for those net circuit segments that must remain fully or partially within the channel area, (nets 5, 6, and 9) in top partition (Figure 3(a)) and (net 7) in bottom partition (Figure 3(b)). In the process it identifies those nets that can be placed on the same track (nets 5,6, and 9) resulting in track savings. It is evident that our algorithm requires only one row (track) to route this channel circuit in each of the partitions, compared to five in [7]. The four net circuits left in the channels of the two partitions are all two terminal nets. Vertical and horizontal segments of nets are placed on the first and second layers respectively in the channel area (VH model) for each 2−layer partition.

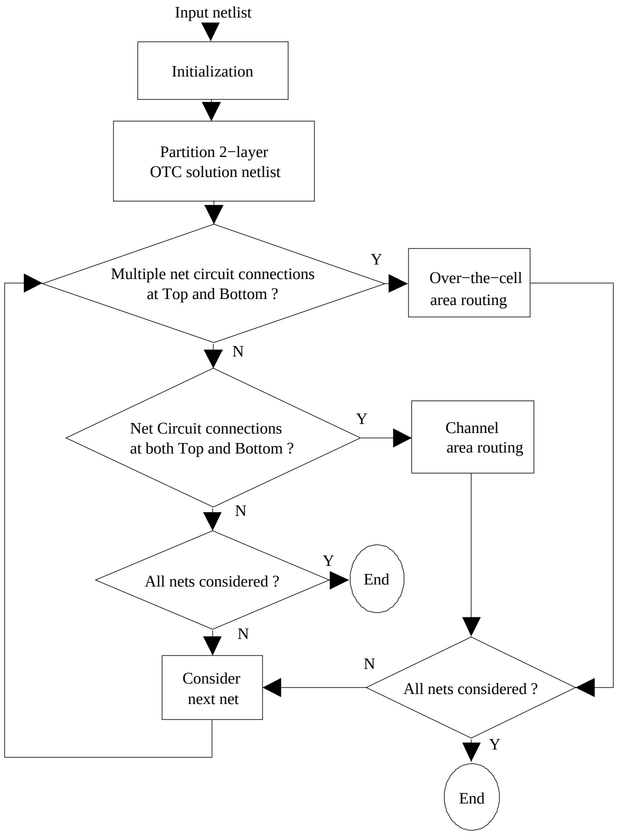

Figure 2: Fitness function flowchart

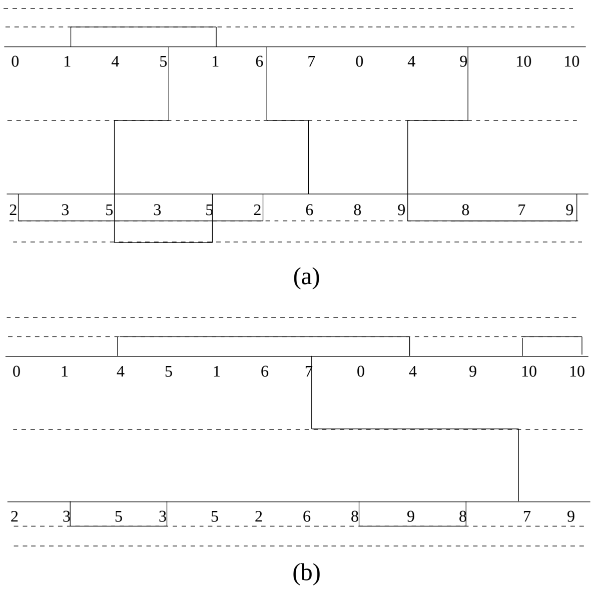

Figure 3: 4−layer solution obtained for YK channel by our algorithm showing (a) Top and (b) bottom 2−layer partitions.

5. RESULTS AND DISCUSSION

Table 1 compares the results of our combined 3D (4−layer) and OTC routing algorithm with those of the best 2−layer over−the−cell router in the literature [4], for all channel areas i.e Top, Bottom and Internal. Overall our algorithm performs better. For example, with Ex3c our algorithm obtains solutions of 6 and 5 tracks for the top and bottom over−the−cell areas respectively, compared to 15 and 16 tracks achieved for the same areas in [4]. Inside the channel our algorithm obtains a solution using 6 tracks compared with 15. Table 2 compares the 4−layer routing results of our algorithm with those of the best multi−layer internal router [2] in the literature when it is using 4−layers also. Here again our algorithm performs better. For example, with example Ex1 our algorithm obtains a solution inside the channel of 3 tracks compared with 8 obtained in [2]. Also, with example Ex3a our algorithm obtains a solution of 4 tracks inside the channel compared with 8 obtained by the algorithm in [2]. In order to determine the extent of the effectiveness of our algorithm, Table 3 compares our 4−layer results with the 5−layer results obtained by the algorithm in [2]. Again our algorithm performs better. For example, with example Ex1 our algorithm obtains a solution with 3 tracks in the internal channel area compared with 6 tracks achieved by the algorithm in [2]. In addition with example Ex3a our algorithm obtains a solution with 4 tracks compared with 5 tracks in [2].

		[4]			*Our Algorithm*		
Example	***Density***	***Top***	***Internal***	***Bottom***	***Top***	***Internal***	***Bottom***
Ex1	12	11	6	7	2	3	3
Ex3a	15	13	15	10	5	4	3
Ex3c	18	15	15	16	6	6	5
Ex4b	17	28	6	24	4	3	6
Ex5	20	24	9	18	5	5	4
De	19	20	13	19	7	7	7

Table 1: Comparison of our algorithm with a well−known algorithm for all areas.

	[2]	*Our*
Example	***Internal***	***Internal***
Ex1	8	3
Ex3a	8	4
Ex3c	9	6
De	10	7

Table 2: Comparison of our algorithm with a multilayer internal router for 4−layers.

	[2]	*Our*
Example	***Internal***	***Internal***
Ex1	6	3
Ex3a	5	4
Ex3c	6	6
De	7	7

Table 3: Comparison of the 4−layer results of our algorithm with the 5−layer results of a multilayer internal router.

6. CONCLUSIONS

A novel genetic algorithm based router is presented for the realization of combined 3D (4−layer) and OTC routing. Given a 2−layer OTC routing solution netlist, the algorithm identifies candidate nets for all areas top, internal and bottom and partitions them into top and bottom 2−layer partitions (sub−problems) in order to minimize area. For each of these partitions the algorithm routes all multi−terminal net circuit segments at the top and bottom parts of the channel over−the−cell in order to minimize channel area, by leaving only two terminal net circuits inside the channel with the effect of reducing congestion. The proposed algorithm achieves significant improvements in routing compared to over−the−cell and multi−layer routers in the literature. When comparison is made with the latest traditional over−the−cell routers, up to 67% more savings is achieved in the channel area. On the other hand, when our algorithm is compared with a multilayer router, up to 62% additional savings in tracks is achieved in the internal channel area. This includes 4−layer results obtained with our algorithm being 50% better than 5−layer results achieved by the best multilayer router in the literature.

REFERENCES

1. MILENKOVICH, A., and MILUTINOVIC, V., A quantitative analysis of wiring lengths in 2D and 3D VLSI, *Microelectronics Journal 29*, Elsevier Science Limited, pp 313−321, 1998.

2. BRAUN, D., BURNS, J., L., ROMEO, F., SANGIOVANNI−VINCENTELLI, A., MAYARAM, K., DEVADAS, S., MA, H., T., Techniques for multilayer channel routing, *IEEE Transactions on Computer−Aided Design*, vol. 7, no. 6, June 1988.

3. CONG, J., WONG, D., F., and LIU, C., L., New approach to three− or four−layer channel routing, *IEEE Transactions on Computer−Aided Design*, vol. 7, no. 10, October 1998.

4. SHIRAISHI, Y., and SAKEMI, Y., A permeation router, *IEEE Transactions Computer−Aided Design*, Vol. CAD−6, pp. 462−471, May 1987.

5. BHINGARDE, S., PANYAM, A., SHERWANI, N., On optimal cell models for over−the−cell routing, *Proceedings of the 6th International Conference on VLSI Design*, Bombay, India, pp. 94−99, January 1993.

6. SZYMANSKI, T. G., Dogleg channel routing is np−complete, *IEEE Transactions on Computer−Aided Design*, CAD−4: 31−41, January 1985.

7. YOSHIMURA, T., and KUH, E., S., Efficient algorithms for channel routing, *IEEE Transactions on Computer−Aided Design of Integrated Circuits and Systems* CAD−1(1):25−35, January, 1982.

8. GOLDBERG, D., E., *Genetic Algorithms in search, Optimization & Machine Learning. Addison−Wesley.* 1989.

9. GOODMAN, E., D., An Introduction to GALOPPS, The "Genetic Algorithm Optimized for Portability and Parallelism" System, Release 3.2, Michigan State University, 1996.

10. HOLLAND, J. H., *Adaptation in natural and Artificial Systems*, MIT Press. 1992.