IMPROVED CHANNEL ROUTING BY VIA MINIMIZATION AND SHIFTING

Chung-Kuan Cheng† and David N. Deutsch

Bellcore
435 South Street
Morristown, N.J. 07960-1961
(201) 829-4335

Abstract

Channel routing area improvement by means of via minimization and via shifting in two dimensions (compaction) is readily achievable. Routing feature area can be minimized by wire straightening. The implementation of algorithms for each of these procedures has produced a solution for Deutsch's Difficult Example, the standard channel routing benchmark, that is more than 5% smaller than the best result published heretofore. Suggestions for possible future work are also given.

Introduction

Two-layer channel routing has advanced considerably since Hashimoto and Stevens [1] first defined the problem and proposed a solution in 1971. In 1973, Kernighan et al. [2] described a branch and bound algorithm that guaranteed "optimality" (minimum track count) for the typical routing style used at that time. In 1976, Deutsch [3] described a heuristic "dogleg" router that achieved results at or near the theoretical lower bound for track count assuming a reserved layer routing strategy (all horizontal segments on one layer, vertical on the other). That paper contained the now famous difficult channel consisting of 72 nets with a density of 19 and a solution using 21 tracks. This example has become the de facto benchmark for channel routing algorithms. Subsequent progress included automatic 20-track solutions by Yoshimura and Kuh [4] and Rivest and Fiduccia [5] and 19-track solutions by Burstein and Pelavin [6] and Sangiovanni-Vincentelli et al. [7]. Since solutions with a track count equal to the channel density had now been achieved, it seemed that future improvement from the standpoint of routing area would be nonexistent except for some fine tuning based on variable track spacing or small contact offsets (reference [8] describes these methods in greater detail).

In 1985, however, Deutsch [8] introduced the notion of compacting traditional channel routing using a contour following procedure and achieved an area savings of 17.6% compared to the best "straight track" solution. The value of this technique was quite evident at the 1987 Design Automation Conference. The technical sessions contained four papers on this subject [9-12] and vendor demonstrations indicated the availability of compacted routing in commercial products.

The data shown in Table 1 indicate channel height versus time for the difficult example using the design rules from [8]. Straight track solutions are based on an edge clearance of 2.1 to the centerline of the first track and a spacing of 2.8 between tracks with contacts offset 0.1. For all of the solutions given here, variable track spacings with contacts centered gave worse results. Note that the paper by Royle et al. [9] claims a height of 44.33 but doesn't state the precise design rules used. In addition, their plot is missing part of one net (it may not affect the solution height), has terminals off grid, and shows other peculiarities. As a result, the last entry in Table 1 may be suspect. A lower bound of 41 has been determined for the vertical compaction of any reserved layer routing based on a density of 19, 20 spacings and two distinct nets requiring contacts (vias) in the maximum density region [8].

Table 1: Channel Routing Progress

Difficult channel height vs. calendar year

Year	Height	Reference
1976	60.2	3
1982	57.4	4,5
1983-4	54.6	6,7
1985	45.0	8
1987	44.3	9

The motivation for the current work was a desire to decrease the channel height still further by means of "intelligent" but simple heuristic algorithms. The crucial phenomenon preventing compaction results from attaining the lower bound is "bump" propagation [8]. Stated another way, the effective width of a via increases as additional nets follow the adjoining contour. This often results in the stacking of vias in the vertical

† Permanent address: Department of Computer Science and Engineering, University of California - San Diego, La Jolla, California 92093-0114.

direction even though the vias have different abscissas. Figure 1 clearly illustrates this effect since the maximal

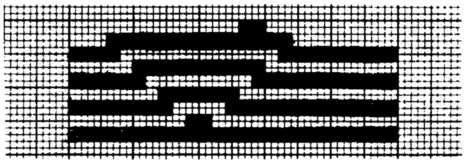

Figure 1: "Bump" Propagation and Via Stacking

height is based upon two paths with vias and two without plus the required separations even though the vias are not vertically aligned. Had the paths with vias been adjacent, however, via stacking would not have occurred. This observation, therefore, leads us to the obvious approaches of trying to minimize the number of vias (fewer bumps and less bump interference) and the notion that lateral (horizontal) via shifting might decrease the stacking described above. Neither of these two ideas is new, as will be seen in the following sections of this paper. Nevertheless, the effective combined use of both simultaneously as well as our wire straightening procedure gives results that are appreciably better than any published heretofore. We also indicate areas for future research.

It is worth noting that our implementation is based on an initial channel routing solution using a reserved layer strategy with two routing layers. We also assume that no net occupies more than one track except when making a dogleg (i.e., no "split nets" in the parlance of [5]). Essentially the same or equivalent techniques can be applied to more general 2-layer solutions and to multilayer (3 or more) solutions. We purposely omit detailed algorithm descriptions due to space limitations.

Via Minimization

The notion that via minimization is worthwhile is certainly not new. In 1976, Deutsch [3] noted that doglegs increased the number of contacts in a routing channel and that it was thus worthwhile to try to minimize the number of doglegs as a secondary figure of merit. Recent references (last year's DAC) include Royle et al. [9] where one of their criteria is the "number of contact windows," Polk [10] where there is a section entitled "Via Reduction" and the paper by Xiong and Kuh [11] where one of their features is "reducing the number of contacts." Via minimization is likely to have a beneficial effect on yield and performance even if we ignore potential area savings. This is even true if a short run of metal is replaced by polysilicon because the capacitance and resistance of the contact are likely to be greater than those of the added poly.

One reason that general 2-dimensional routers and non-reserved layer channel routers obtain better solutions than traditional reserved layer routers is because their solution space is bigger. By using postprocessing steps of via minimization and vertical compaction, we convert a reserved layer routing result into a solution that contains horizontal and vertical features on each routing layer.

Our initial routing solution is based on a reserved layer dogleg channel router where no net exists on two distinct tracks except at dogleg positions. Vias can be eliminated whenever a segment has vias at both ends and the segment is not crossed by a perpendicular segment on the other mask level. For the vertical layer, this situation typically occurs when a dogleg results in a track change but there are no other horizontal features on any intervening track at this position. The ability to minimize vias on the horizontal layer is usually much more common and depends on nets having terminals on adjacent or nearby grid positions. If one of the routing layers has far superior electrical characteristics, it is easy to limit the via minimization procedure to segments whose length is less than some specific amount.

The via minimization procedure is done once as the initial postprocessing step. It consists of two separate passes, one for the horizontal layer, the other for the vertical. Each segment is examined based on the above criterion for via elimination. Successful candidates have their mask layer changed or are replicated on the other mask layer depending upon the precise routing situation. We can think of no reason why such postprocessing should not be done for all routing output, since it improves chip performance and yield.

This heuristic does not guarantee a minimum number of vias. It is easy to create routing configurations where vias can be saved but our procedure is ineffective. Fortunately, such situations occur infrequently in real examples.

Via Shifting

Via shifting (or "offset contacts") has been used for at least 10 years in the LTX layout system [13, 14], although this technique was not described in the original papers. The compaction of [8] and subsequent work [9-12] make significant area savings based on vertical via shifting. Xiong and Kuh [11] indicated that they were enhancing Nutcracker "to allow contacts to slide in both directions." Our approach here is quite similar.

We only give a brief overview of the procedure. In essence, our goal is to determine some lateral (horizontal) via shifts that minimize the bump propagation/stacking problem yet do not change the effective routing topology on either layer. Just as the original compaction produces short vertical segments on the horizontal layer, lateral via shifting produces short horizontal segments on the vertical routing layer.

After via minimization, we perform standard vertical compaction as described in [8]. Of course, we do this compaction for each routing layer. Vias are considered as belonging to both layers. During this compaction step, we build lists of bump propagation chains that indicate the cause of the height for each horizontal segment. Note that a bump is usually caused by a via but it may also be a net segment end if a via as been eliminated from that position. We focus our attention on the features with the greatest height.

We partially straighten the compacted routing (see next section for details) to spread the white space. We then do two separate compactions, one to the left and one to the right, to determine the feasible range for horizontal contact shifts. Knowledge of the critical chains indicates the candidate vias for shifting. We select a small set of vias, compute the minimum shifts needed to reduce or eliminate stacking, and recompact. Our figure of merit is height first, length of features on all mask levels requiring the maximal height second. We iterate this series of steps until no further improvements are made. Finally, we "fully" straighten our best result.

Wire Straightening

Although wire straightening does not directly decrease the channel area, it is still a necessity for our approach to work. In a completely compacted routing environment, lateral via shifting would be essentially impossible because there would be no empty space where such shifts could occur. Of course, straightening of the final result also decreases path lengths, routing capacitance and resistance, and increases circuit speed and yield.

Our straightening, as expected, is done in the reverse order of net compaction. Initially, all nets are compacted down as far as possible. For a rectangular channel, nets in the top track could be made perfectly straight by placing all such routing one design rule separation below the channel top ("ceiling"). Straightening is then applied to nets from the next highest track, etc. Each part of such a net starts as low as possible ("floor"). We compute the current ceiling based on the nets from higher tracks that have already been straightened. The current net is then straightened as much as possible within the area defined by the floor and the ceiling (see reference [9] for a similar discussion).

As noted in the previous section on via shifting, we use two different straightening procedures. One is partial, the other is full. The distinction lies in the initialization phase of the ceiling; otherwise, the algorithms are identical. In the partial situation, the starting ceiling is not a straight line. Instead, we shape the ceiling such that it permits the maximum compacted height only in the interval(s) that require it. All other intervals have the ceiling set one unit lower. This procedure drastically increases the likelihood that subsequent via shifts will be productive or neutral, since lateral shifts will not make use of this area. Full straightening is used only when the final result is produced, since no further modifications will be attempted.

Results and Discussion

We have implemented the via minimization, via shifting and wire straightening algorithms in a C language program running on a VAX 8650 under the Ultrix operating system. This program also runs on a Sun 3 workstation. Although the software is still being debugged, we can report the following preliminary results.

More than 100 distinct solutions to Deutsch's Difficult Example using a "dogleg" router [3] and 10 without doglegs have been generated. Each solution was then processed in order to determine the effectiveness of the above techniques. Program output consisted of the final channel height for each of the following procedures:

1. Horizontal feature compaction [8] only
2. Via minimization and horizontal feature compaction
3. Lateral via shifting and horizontal feature compaction
4. Via minimization, lateral via shifting and horizontal feature compaction

In general, solutions without doglegs yield lower channel heights. This is to be expected since they have fewer vias initially. The average final height for minimum via solutions was approximately 45 while for those with doglegs, it was about 47.3. It is also obvious that each of the techniques contributes to the overall height decrease.

The following verified results are based on the 19-track 9-dogleg solution mentioned in [8]. Applying via minimization followed by straight vertical compaction produced a channel height of 44 rather than the 45 reported by Deutsch [8]. Use of lateral via shifting without via minimization also produced a result of 44. Application of both algorithms yielded a height of 42 which is more than 5% smaller than the best published solution [9]. Compacted routing progress is summarized in Table 2. Our result with a height of 42 is shown in Figure 2. We purposely do not use slant lines because the likely additional savings are insignificant. Slant lines also complicate mask checking routines and may be unacceptable to certain foundries.

Table 2: Compacted Routing Progress

Difficult channel height vs. calendar year, techniques used

Year	Height	Reference	Techniques used
1985	45.0	8	horiz. feature compaction (h.f.c.)
1987	44.3	9	varied
1988	44.0	this work	via minimization (v.m.), h.f.c.
1988	44.0	this work	lateral via shifting (l.v.s.), h.f.c.
1988	42.0	this work	v.m., l.v.s. and h.f.c.

Our result is less than 2.5% above the lower bound given in [8] but, in fact, the actual lower bound has decreased to 40 because of the via minimization and shifting. We do not know whether the lower bound or the solution height can be decreased further. Compared to the complete straightened solution generated in [8], our current result is 6.7% smaller, has 8% fewer contacts, and approximately 3% less length for the vertical routing layer. The horizontal layer has 2.6% more length than the original uncompacted 19-track solution.

Summary and Future Work

We have shown the effectiveness of via minimization and shifting in reducing channel routing area. Our best result for Deutsch's Difficult Example is more than 5% smaller than the best result published heretofore. The same techniques can be applied to nonrectangular channels, multilayer (3 or more) routing, variable width routing paths, and nonreserved layer routing. More general two-dimensional compaction schemes, or the use of slant lines, or both, could probably save slightly more area but would increase the complexity of the procedures. We doubt that such efforts are justified for this special problem. Applying our algorithms to a few routing solutions and accepting the best one would probably be as effective as a more complicated method applied to a single solution (utilizing a comparable amount of computer resources). Since our results indicate a considerable variability in final height for different initial routing solutions, additional work should be focused on generating "good" starting solutions. It might also prove worthwhile to look for local topological changes that would allow additional via reduction.

References

[1] A. Hashimoto and J. Stevens, "Wire Routing by Optimizing Channel Assignment within Large Apertures," Proc. 8th Design Automation Workshop, pp. 155-169 (1971).

[2] B.W. Kernighan, D.G. Schweikert and G. Persky, "An Optimum Channel-Routing Algorithm for Polycell Layouts of Intergrated Circuits," Proc. 10th Design Automation Workshop, pp. 50-59 (1973).

[3] D.N. Deutsch, "A 'Dogleg' Channel Router," Proc. 13th Design Automation Conference, pp. 425-433 (1976).

[4] T. Yoshimura and E.S. Kuh, "Efficient Algorithms for Channel Routing," IEEE Trans. on CAD of ICs and Systems, Vol. CAD-1, No. 1, pp. 25-35 (1982).

[5] R.L. Rivest and C.M. Fiduccia, "A 'Greedy' Channel Router," Proc. 19th Design Automation Conference, pp. 418-424 (1982).

[6] M. Burstein and R. Pelavin, "Hierarchical Channel Router," Proc. 20th Design Automation Conference, pp. 591-597 (1983).

[7] A. Sangiovanni-Vincentelli, M. Santomauro and J. Reed, "A New Gridless Channel Router: Yet Another Channel Router the Second (YACR-II)," Proc. of the IEEE International Conference on Computer-Aided Design, ICCAD-84, pp. 72-75 (1984).

[8] D.N. Deutsch, "Compacted Channel Routing," Proc. of the IEEE International Conference on Computer-Aided Design, ICCAD-85, pp. 223-225 (1985).

[9] J. Royle, M. Palczewski, H. VerHeyen, N. Naccache, and J. Soukup, "Geometrical Compaction in One Dimension for Channel Routing," Proc. 24th Design Automation Conference, pp. 140-145 (1987).

[10] D.B. Polk, "A Three-Layer Gridless Channel Router with Compaction," Proc. 24th Design Automation Conference, pp. 146-151 (1987).

[11] X.-M. Xiong and E.S. Kuh, "Nutcracker: An Efficient and Intelligent Channel Spacer," Proc. 24th Design Automation Conference, pp. 298-304 (1987).

[12] C.H. Ng, "A 'Gridless' Variable-Width Channel Router for Macro Cell Design," Proc. 24th Design Automation Conference, pp. 633-636 (1987).

[13] G. Persky, D.N. Deutsch and D.G. Schweikert, "LTX - A System for the Directed Automatic Design of LSI Circuits," Proc. 13th Design Automation Conference, pp. 399-407 (1976).

[14] G. Persky, D.N. Deutsch and D.G. Schweikert, "LTX - A Minicomputer-Based System for Automated LSI Layout," Journal of Design Automation & Fault-Tolerant Computing, Vol. 1, No. 3 (May 1977), pp. 217-255.

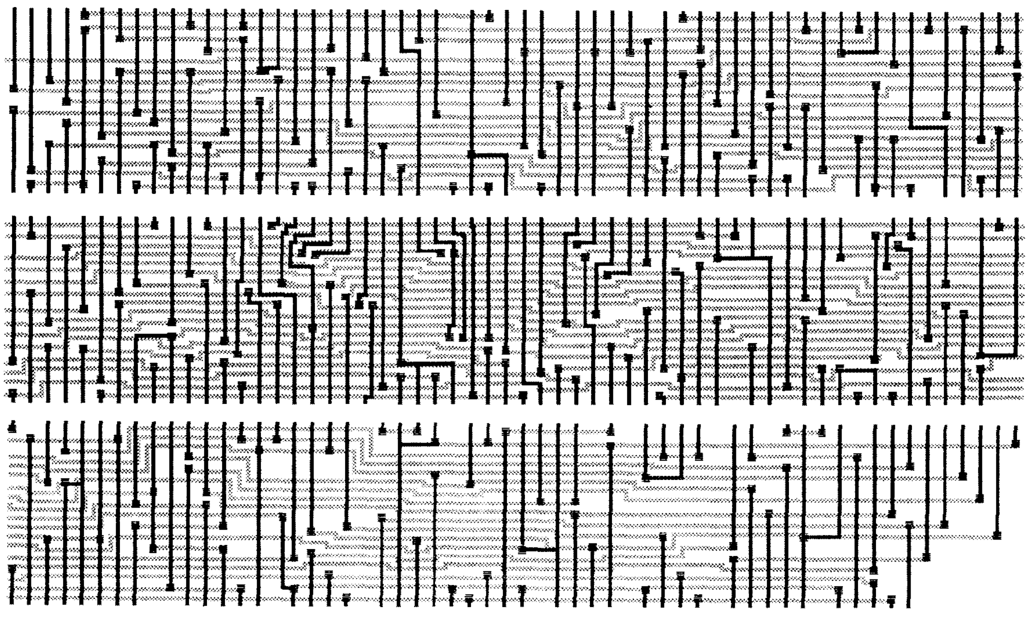

Figure 2: Deutsch's Difficult Example with a height of 42
Top: Left edge to column 58; Middle: columns 59-116; Bottom: columns 117-174