Effective and Efficient Detailed Routing with Adaptive Rip-up Scheme and Pin Access Refinement

Zhongdong Qi*
Xidian University
Xi'an, China
zdqi@xidian.edu.cn

Jinchong Zhang
Giga Design Automation
Shenzhen, China
jczhang@giga-da.com

Gengjie Chen
The Chinese University of Hong Kong
Hong Kong SAR, China
gjchen@cse.cuhk.edu.hk

Hailong You
Xidian University
Xi'an, China
hlyou@xidian.edu.cn

ABSTRACT

Detailed routing is one of the most complex and time-consuming stages of VLSI design process. Due to the rapidly growing problem scale and increasing number of design rules in advanced technology nodes, a feasible routing result can only be achieved after many rounds of rip-up and reroute (R&R) iterations, which takes a significantly long runtime. In this paper, we propose several effective and efficient techniques to handle the design rule violations in detailed routing. An adaptive rip-up scheme with two strategies of different effort is designed, which can speed up the R&R phase with comparable solution quality. To cope with the pin access challenge with complex design rule constraints, approaches to refine the pin connections are proposed. Besides, some specific design rules are handled in a post-processing manner efficiently. Experiment result shows that the number of design rule violations can be reduced by 69% with 28% lower runtime on average, after integrating these techniques in Dr. CU 2.0.

CCS CONCEPTS

• **Hardware → Wire routing**.

KEYWORDS

Detailed routing, physical design, rip-up and reroute, design rules, pin access

ACM Reference Format:
Zhongdong Qi, Jinchong Zhang, Gengjie Chen, and Hailong You. 2022. Effective and Efficient Detailed Routing with Adaptive Rip-up Scheme and Pin Access Refinement. In *Proceedings of the Great Lakes Symposium on VLSI 2022 (GLSVLSI '22), June 6–8, 2022, Irvine, CA, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3526241.3530361

*Corresponding author

GLSVLSI '22, June 6–8, 2022, Irvine, CA, USA

ACM ISBN 978-1-4503-9322-5/22/06…$15.00
https://doi.org/10.1145/3526241.3530361

1 INTRODUCTION

Detailed routing is one of the most complex and time-consuming stages in VLSI design process, which searches interconnections and constructs wires and vias. It handles different kinds of design rules introduced by successive technology nodes [1], while minimizing the wirelength and via counts. With manufacture technology advances, the larger scale of designs and the increasing complexity of design rules make the detailed routing task even more challenging. Therefore, effective and efficient techniques for detailed routing is of great importance to reduce the turn-around time in VLSI design.

There are two major categories of detailed routing approach in previous works. One is sequential rip-up and reroute (R&R) (e.g., BonnRouteDR [2]). The other is concurrent optimization using multi-commodity flow or integer linear programming (e.g., MCFRoute [3]). Prompted by the ISPD initial detailed routing contests [4, 5], several high quality academic detailed routers were developed recently (e.g., [6–10]). In these works, the sequential approach using iterative R&R gained more popularity.

In sequential detailed routing, the R&R phase typically takes a large amount of time. Several rounds of R&R iterations are needed to remove design rule check (DRC) violations. It is essential to reduce the runtime of the R&R phase for efficient detailed routing. In addition, although path search algorithms can handle important design rules [7, 10, 11], effective post-processing is needed to fix remaining DRC violations. Moreover, because of complex pin shapes and related design rules, pin access connections need to be designed carefully to prevent DRC violations.

In this paper, we propose some effective and efficient techniques in sequential detailed routing, to reduce runtime and the number of DRC violations. These techniques are compatible with previous on-grid sequential detailed routers. By applying these techniques in an academic detailed router Dr. CU 2.0 [8], the number of DRC violations and the detailed routing runtime can be reduced by 69% and 28% on average, respectively, with comparable wirelength and via count. Main contributions of this work are as follows.

- An adaptive rip-up scheme that achieves significant reduction of runtime in R&R phase with comparable quality of results.
- Pin access refinement techniques including macro pin access point assignment and pin connection geometry refinement, which effectively reduce the number of DRC violations.

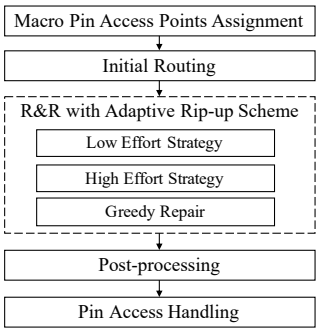

Figure 1: Overall detailed routing flow.

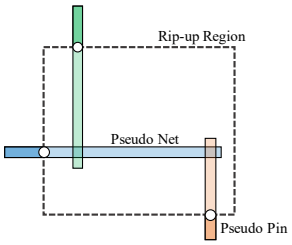

Figure 2: A pseudo net with pseudo pins in a rip-up region.

- Efficient post-processing techniques to handle important design rules that are difficult to satisfy during path search.

2 METHODOLOGY

The detailed routing flow with our proposed techniques is shown in Fig. 1. First, proper access points are assigned to all macro pins. Then initial routes are constructed. The following R&R phase runs with the adaptive rip-up scheme for high efficiency. Violations of some important design rules are fixed by post-processing after R&R. Finally, the quality of pin connections is improved by via shifting and area supplementing techniques.

2.1 Adaptive Rip-up Scheme

The R&R phase is crucial for repairing DRC violations after initial routing. The runtime and result quality of the R&R phase depend on the rip-up strategy, because 1) the rerouting region is generally affected by the boundary of removed paths, and 2) the possible new paths constructed during rerouting are restricted by the remaining paths after rip-up. To achieve good solution quality in short runtime, an adaptive rip-up scheme is proposed. It can adjust the repairing effort by adapting the rip-up region according to different level of DRC violations. The repairing effort can be controlled by assigning rip-up regions with proper size, in which the interconnection paths (wires and vias) should be removed. The consecutively removed wires and vias of a net are regarded as a pseudo net, and grids at which they are connected with the remaining path are pseudo pins, as shown in Fig. 2.

There are three types of R&R with adaptive rip-up scheme. The first type is a low effort rip-up strategy which assigns rip-up regions with a small, fixed size. It is particularly fast. The purpose of the low effort repair is to quickly fix a large amount of DRC violations which do not have high routing space demand. The second type is a rip-up strategy with higher repair effort, which customizes the rip-up region of each violation respectively. The last type is greedy optimization which ignores the history cost of previous DRC violations in rerouting and tries to find better routing paths.

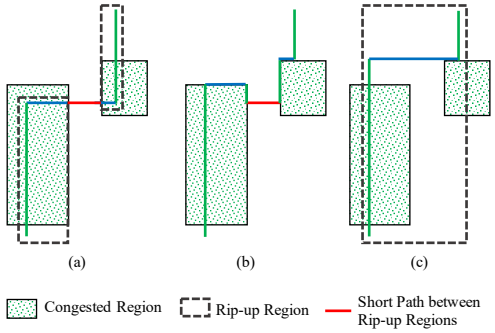

Figure 3: Removing the short path can avoid unnecessary detour: (a) A short path between two rip-up regions; (b) Rerouting result with the short path remained; (c) Rerouting result after merging rip-up regions.

To rip-up and reroute a net, the coordinates of wires or vias with DRC violations in grid graph are collected, and rip-up regions are assigned according to different strategies. Then the rip-up regions are merged if they are too close to each other. After that, the detailed router constructs pseudo nets, and calls the path search engine to reroute these pseudo nets.

2.1.1 Merge Rip-up Regions. Before constructing the pseudo nets, the rip-up regions need to be checked whether some of them are too close each other. As shown in Fig. 3, the detours are inevitable when rerouting the pseudo nets using the rip-up regions defined in Fig. 3(a). They can be avoided by merging the two rip-up regions if the short path between these two rip-up regions is also removed.

2.1.2 Low Effort Strategy. In order to quickly obtain an initial result, detailed routers typically limit the computational effort during the initial routing phase by relaxing violation penalties or using relatively small routing regions. As a result, after initial routing, many DRC violations have sufficient space around, and thus can be easily fixed locally. Therefore, we only remove the wires and vias in small local regions, which are represented by rectangular boxes in the grid graph with a violation at each box's center.

2.1.3 High Effort Strategy. Different from the low effort one, the high effort strategy aims to fix DRC violations especially in congested regions. The rip-up region here is not merely a box with a fixed size used in the low effect strategy. Since the volume of available routing region around the congested regions varies, it is inefficient to use a fixed size rip-up region to cope with all violations. Even if the rip-up region contains adequate empty track segments, it may result in undesired detours, as shown in Fig. 4 (a). Therefore, the rip-up region in high effort strategy is adjusted for each DRC violation considering both congestion and the net's topology.

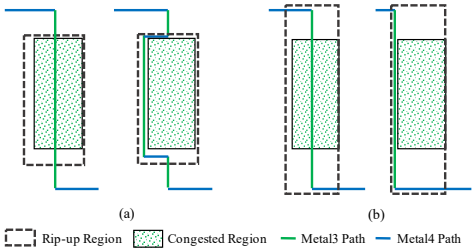

Figure 4: Assign rip-up region with consideration of net topology: (a) An inappropriate rip-up region results in a detour; (b) A proper rip-up region includes Metal4 path for track switching.

In high effort strategy, the rip-up region is selected by using the global route rectangles containing DRC violations, as shown in Fig. 4 (b). It uses the topology of the net, which results in a larger rerouting space to fix DRC violations and avoid unnecessary detours.

2.2 Pin Access Assignment and Refinement

As one of the most difficult problems in detailed routing, pin access has many complex constraints and situations. We proposed some approaches to tackle pin access issues. For macro pins, we assign proper access points to cope with the macro blockages. Generally, for macro pins and standard cell pins, a pin via shifting technique is designed, to fix the wide metal spacing violations. We also ensure sufficient overlap between pin metals and routed paths, by area supplementing in pin connections.

2.2.1 Macro Pin Access Point Assignment. As shown in Fig. 5 (a), macros are typically blocked by routing blockages on multiple layers to protect the internal connections. The space around the macros is also blocked due to metal layer spacing rules. A macro pins can only be accessed by wires perpendicular to the pin boundary on the macro pin metal layer(s), without overlapping with routing blockages around the pin. For example, the wire connection in Fig. 5 (a) is invalid because it overlaps with the routing blockage, while the one in Fig. 5 (b) is valid.

To cope with the situation and constraints, we define a safe region for macro pin connections, as shown in Fig. 5 (b). The safe region is selected according to macro pin position, the routing blockages around the macro pin, and spacing rules. The grid points near the boundary of the safe region and inside the macro pin are valid access points, and the ones in the safe region but not inside the macro pin are extra access points. The other grid points inside the macro pin are marked as invalid access points, with high penalty in path search. This results in pin connections accessing the macro pin in the safe region.

2.2.2 Pin Connection Via Shifting. To ensure lithography and manufacturability, a spacing larger than the normal spacing is required between a wide metal and other geometries near it. Wide metal spacing violations mostly occur when connecting to pins on low metal layers, due to high metal density and irregular shapes of pins. As shown in Fig. 6 (a), the via bottom enclosure, pin shape

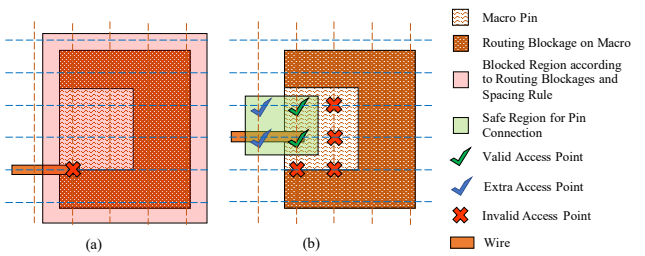

Figure 5: Assign valid access points for macro pin: (a) Invalid pin access wire which overlaps with routing blockages; (b) Valid access points and safe region for pin connection.

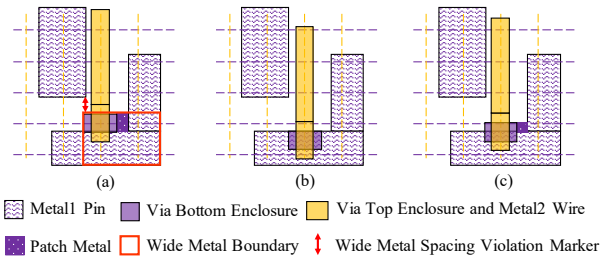

Figure 6: Fixing wide metal spacing violation by pin via shifting: (a) A spacing violation between a pin and a wide metal; (b) Shifting the pin via to make it be covered by the pin metal; (c) Shifting the pin via with precise distance to fix wide metal spacing.

and patch shape for fixing normal spacing violation form a wide metal shape with a larger spacing requirement, which results in a violation between the other pin shape. To fix the violation, we find legal off-grid access points for pin vias by a two-stage shifting, as shown in Fig. 6 (b) and Fig. 6 (c). Because the via can be shifted towards the pin metal, and the shifting distance is usually small, it is capable to fix most of these violations.

2.2.3 Pin Connection Area Supplementing. For the intersection rectangle of two metals, the sufficient overlap rule requires its width to be no less than a minimum metal width on current layer. Violations of this rule usually occurs on pin connections, because most of the pin metals are off grid. We extend affected pin access path by a small wire segment or a L-shape wire to ensure sufficient overlap.

2.3 Design Rule Specific Postprocessing

In detailed routing, most of the design rules are satisfied after the R&R phase. There are still some rules which are difficult to check during path search, but can be efficiently handled by post-processing, because the violations can be fixed locally. We propose two effective post-processing techniques to fix the violations of cut spacing rule and min area rule, respectively.

2.3.1 Cut Spacing Rule. Once a new via is used, taking the cost of cut spacing rule into account require back-tracing to find the former via. However, to sufficiently explore the routing space, a

Table 1: Comparison of final results between Dr. CU 2.0 [8] (CU) and the one with our proposed techniques (Ours)

Testcase	CU				Ours			
	WL (μm)	#Via	#DRV	Time (s)	WL (μm)	#Via	#DRV	Time (s)
test1	86720	32394	1723	7.93	87030	32433	6	6.17
test2	1567000	325674	20093	80.59	1568000	326176	24	65.64
test3	1744000	318303	21549	123.37	1745000	319566	167	81.84
test4	2642000	729197	978	624.92	2651000	747635	308	283.36
test5	2780000	965546	778	426.57	2779000	969788	66	277.63
test6	3570000	1480728	716	512.95	3569000	1488168	123	431.99
test7	6517000	2402487	962	1032.17	6518000	2416205	188	762.37
test8	6547000	2412163	984	1016.33	6546000	2425808	164	712.46
test9	5476000	2410625	337	829.42	5476000	2421202	125	714.71
test10	6810000	2595706	4605	1640.95	6813000	2612279	8080	1135.85
Avg. Ratio	1.00	1.00	1.00	1.00	1.00	1.00	0.31	0.72

great number of paths would be tried by a router. Each route may include many vias. Therefore, detecting the violations of vias in the same net during path search is quite inefficient. To check cut spacing rule, we extract each net's vias, record their coordinates in grid graph, and check if there are other vias on the surrounding grids. For via cuts violating cut spacing rule, we try to move them to neighboring positions to fix the violations.

2.3.2 Minimum Area Rule. It is most common for via enclosures to violate the minimum area rule. Considering there are many via types for via selection, and there may be a redundant via insertion choice during path search, it is inefficient and sometimes difficult to check min area rule in path search. As a result, we use a post-processing approach to fix min area rule violations. We get all the separated metals of a net by a breadth-first search of its routed paths and calculate their precise area. For each metal that violates the min area constraint, we query the violation-free intervals on the tracks it passes and add patch metals if it is possible to fix.

3 EXPERIMENTAL RESULTS

The proposed techniques are implemented in C++ and integrated into Dr. CU 2.0 [8]. The benchmarks we used are ISPD 2018 initial detailed routing benchmark suite [4]. We perform the experiments on a Linux machine with an 8-core Ryzen CPU running at 3.2GHz. Examinations of wirelength, via count and number of design rule violations (denoted as #DRV in this section) of the routing result are done by Cadence Innovus 15.2.

To evaluate the overall performance and solution quality, the statistics of detailed routing results using all these techniques are also listed in Table 1. In the table, CU and Ours are both full flow detailed routing. We can see that by using the proposed techniques, the number of design rule violations and the detailed routing runtime are reduced by 69% and 28% on average respectively, with similar wirelength and via count. This demonstrates the effectiveness and efficiency of these techniques.

4 CONCLUSIONS

In this paper, we propose some effective and efficient techniques for detailed routing, to tackle the challenges in advanced technology nodes. We propose an adaptive rip-up scheme that is capable to significantly speed up the R&R phase with comparable result quality. Several pin access refinement approaches are proposed, which can effectively reduce violations of different design rules related to pin connections. In addition, important design rules including cut spacing and min area are effectively handled by post-processing techniques.

ACKNOWLEDGMENTS

The authors would also like to thank the anonymous referees for their valuable comments and helpful suggestions. This work was supported by the National 111 Center. This work was also partially supported by the Fundamental Research Funds for the Central Universities.

REFERENCES

[1] A. B. Kahng, J. Lienig, I. L. Markov and J. Hu, "VLSI physical design: From graph partitioning to timing closure," Springer, 2011.
[2] M. Ahrens, M. Gester, N. Klewinghaus, D. Müller, S. Peyer, C. Schulte and G. Téllez, "Detailed routing algorithms for advanced technology nodes," in *IEEE Trans. on CAD* 34(4) (2015), pp. 563–576.
[3] X. Jia, Y. Cai, Q. Zhou, G. Chen, Z. Li and Z. Li, "MCFRoute: A detailed router based on multi-commodity flow method," in *Proc. ICCAD*, 2014, pp. 397–404.
[4] S. Mantik, G. Posser, W.-K. Chow, Y. Ding and W.-H. Liu, "ISPD 2018 initial detailed routing contest and benchmarks," in *Proc. ISPD*, 2018, pp. 140–143.
[5] W.-H. Liu, S. Mantik, W.-K. Chow, Y. Ding, A. Farshidi and G. Posser, "ISPD 2019 initial detailed routing contest and benchmark with advanced routing rules," in *Proc. ISPD*, 2019, pp. 147–151.
[6] F.-K. Sun, H. Chen, C.-Y. Chen, C.-H. Hsu and Y.-W. Chang, "A multithreaded initial detailed routing algorithm considering global routing guides," in *Proc. ICCAD*, 2018, pp. 82:1–82:7.
[7] G. Chen, C.-W. Pui, H. Li, and E. F. Y. Young, "Dr. CU: Detailed routing by sparse grid graph and minimum-area-captured path search," in *IEEE Trans. on CAD* 39(9) (2020), pp. 1902–1915.
[8] H. Li, G. Chen, B. Jiang, J. Chen and E. F. Y. Young, "Dr. CU 2.0: A scalable detailed routing framework with correct-by-construction design rule satisfaction," in *Proc. ICCAD*, 2019, pp. 1–7.
[9] A. B. Kahng, L. Wang and B. Xu, "TritonRoute: The open-source detailed router," in *IEEE Trans. on CAD* 40(3) (2021), pp. 547–559.
[10] S. M. M. Gonçalves, L. S. Rosa and F. S. Marques, "DRAPS: A design rule aware path search algorithm for detailed routing," in *IEEE Trans. on CAS II* 67(7) (2020), pp. 1239–1243.
[11] F.-Y. Chang, R.-S. Tsay, W.-K. Mak and S.-H Chen, "MANA: A shortest path maze algorithm under separation and minimum length nanometer rules," in *IEEE Trans. on CAD* 32(10) (2013), pp. 1557–1568.