Fusion of Multidisciplinary Research, An International Journal (FMR) , Vol. **7**, Iss. **1**, 2026
https://doi.org/10.63995/YSQW8834

FMR

ARTICLE

Macro-Aware Keepout Margin Optimization for Timing and Routability

Polisety Omkar,[1] Bantupalli Ranganath,[1] Srinidhi Prabhakar,[2] and Vallabhuni Vijay [*,2]

[1]Silica Launch, Bangalore, India
[2]AstraSilica Technologies, Bangalore, India
*Corresponding author: vvijay@astrasilica.com

(Received: 21 August 2025; Revised: 09 November 2025; Accepted: 14 December 2025; Published: 01 January 2026)

Abstract

The strategic integration of Intellectual Property (IP) macros between soft IPs and hard IPs is the base framework in the attainment of design timing closure and improved performance. Since the main limitations to signal timing, power distribution and silicon area efficiency are determined by these high-density blocks, their location is a critical factor in determining chip performance. Another major drawback of the traditional workflows is the physical integration that is not optimal, namely the definition of keepout margins. Small keepout margins around macros can easily lead to routing congestion where too much standard cell density around macro boundaries causes a lot of congestion. To reduce this, macro placement applies flyline analysis to chart logical connectivity, and avoids congestion by making sure the orientation is proper, keepout margins are mathematically sized to fit all the tracks, and has a huge impact on the QoR and the timeline to tape-out.

Keywords: Congestion Analysis; Flyline Analysis; ICC2; Keepout Margin; Macro Placement; Physical Design; Routability; Timing Closure

Abbreviations: ASIC: Application-Specific Integrated Circuit, CTS: Clock Tree Synthesis, IP: Intellectual Property, NDR: Non-Default Rule, PPA: Power, Performance, and Area, QoR: Quality of Results, SoC: System on Chip, TNS: Total Negative Slack, WNS: Worst Negative Slack

1. Introduction

Current ASIC floorplanning is the architectural foundation of physical design, and is heavily based on the strategic placement of both Soft and Hard IP macros in meeting aggressive timing closure and high-performance goals. These silicon-proven, high-density blocks serve as the main physical objects, which determine the key limitations to global power distribution systems and the overall efficiency of silicon area in general. Since macros take up a large portion of the core area, their spatial location is the determining factor of the success of a chip, and it directly affects the distribution of the wires and the signal latency. The conventional physical design processes usually face serious bottlenecks in integration, which may include routing congestion, signal integrity loss. In this paper, the authors discuss the application of precision-driven macro arrangement by advanced connectivity analysis and orientation optimization to successfully pass these challenges and improve the Quality of Results (QoR) [1].

1.1 Importance of Macro Integration in ASIC Floorplanning

The most important aspect that determines whether a layout of a chip will be successful or not is the macro placement, which has a direct impact on the speed of the signal and the congestion of the wires. These high-density blocks are large and form the skeleton of the chip and determine the distribution of signals and power within the silicon. In current multifaceted styles, the good of this primary set-up is the difference between the design being able to achieve performance goals or not. Strategic integration also makes sure that those blocks do not interfere with crucial communication channels and offer a consistent base that minimizes the necessity to implement new designs and changes all the time [1].

1.2 Limitations of Fixed Keepout Margins

One of the main problems of conventional planning is the application of hard, fast keepout margins, surrounding these big blocks. At high-density designs like designs with more than 500k components and high space utilization, congestion occurs in the routing process whereby the wires are overcrowded and cannot be connected. On the other hand, too large margins are costly in terms of silicon space. The issue is also enhanced by the shapes of the chips which include rectangular and rectilinear boundaries. This paper discusses ways of going beyond the zero margins to come up with the optimal balance to ensure that the congestion is not more than 0.1% and yet at the same time the area is efficient enough [2].

1.3 Impact of Macro Flylines on Routability

In order to make sure that wires can be successfully connected, designers resort to flyline analysis in order to visualize the way in which various blocks are logically interconnected. This analysis is useful in identifying the most appropriate orientation and position of each block in order to ensure that pins are exposed to the most common connection points. With these visual aids, designers are able to draw accurate keepout areas that allow sufficient space between signals in order to pass between the inner layers of a block and the primary routing grid. This reduces white space and makes signal paths as short as possible, which is necessary to ensure speed and reliability of the chip [2].

1.4 Effectiveness of This Approach

The paper offers a methodical manner of automating the planning of the many shapes of chips and design needs and thus saves a lot of time in the development cycle. This work determines the most effective approaches to timing, power, and congestion management in both rectangular and non-rectangular chip designs by experimenting with the placement strategies on high-density designs. The findings indicate that the time to spend on layout can be cut down to days by selecting an appropriate placement method to use in a particular chip shape. It is a methodology that provides a clear way to attain quality results and has a quicker way to the final production stage.

2. Design Overview

The adaptive keepout block is a high-density integration system that is used to test the repeater insertion to optimize timing methodology. It operates as a time remedial critical timing block in which long-distance global nets are buffered to achieve signal steepness and achieve aggressive clock frequency targets. The design passes through an entire physical implementation cycle, which begins with a gate-level netlist and ends with a DRC/LVS clean GDSII. This block is particularly defined by the fact that it is highly dependent on IP macros, so it is a good fit in terms of investigating the influence of the keepout margins on the overall routing congestion and silicon usage.

2.1 Design Parameters: Adaptive keepout

The adaptive keepout block is a high density, timing critical subsystem that is built in 28nm CMOS that is designed expressly to support repeater insertion to optimize timing. The structure controls about 75,000 leaf cells in an area of 313,567mm^2. It is characterized by a huge macro-dominance: six hard memory macros cover 250,205mm^2, which is an 80 percent macro-to-core area ratio. The block is required to address the serious latency problems in two global nets with fanouts of over 1,000 at a target frequency of 1.2 GHz. This tight floorplan is a highly dense constrained environment in which repeaters with high drive have to be carefully positioned in the tiny 20% whitespace. The flow of implementation, between RTL and a DRC-clean GDSII, is based on the optimisation of keepout that is accurate at the macro-level to avoid routing saturation in the straight channels between dominant IP blocks.

2.2 Technology and Design Scale

It uses the 28nm technology node that involves tight control of power density and signal interference. The geometry used at this geometry requires the application of Non-Default Rules (NDRs) to avoid the electromigration and crosstalk of high-frequency nets. The technology option enables the exploration of Power, Area and Performance (PPA) trade off in high density environment. This node presents a realistic model of testing the effect of macro placement in routing congestion and manufacturing feasibility [3].

2.3 Macro Characteristics and Placement Constraints

The placement in the adaptive keepout block is done using rigorous flyline analysis to ensure alignment of physical coordinates against logical data flow to in effect reduce global wirelength. These dense blocks are carefully placed so that the pin access is not hindered and thus the possibility of routing bottlenecks and routing congestion at the macro boundaries is reduced. The design achieves a very tight trade-off between macro proximity and routing track availability by placing very strict limits on placement and keepout areas [4]. This reduces the amount of white space and is the highest density of core, which guarantees that the signal paths are straight and efficient. This precision is essential in terms of timing closure as well as signal integrity throughout the 28nm fabric [4].

2.4 Keepout Configuration

Keepout margins are added to the macro boundaries as technically specified exclusion areas to ensure that the standard cells do not hinder the access of pins. These margins are special routing paths that are used to allow signals to pass on to the global routing grid without causing hotspots of congestion. These keepouts should be sized properly to prevent the problem of "routing congestion," which offers the mathematical trade-off between high core density and the physical routability of the resulting layout [4].

3. Keepout Margin and Flyline Induced Routing Challenges

Modern macro-dominant ASIC design is typically confronted by a severe routing congestion and timing uncertainty due to early choices during floorplanning. The standard and identical keepout margin around macros is not sensitive to the variations in pin density and logical connectivity that cause inefficient use of routing resources and open up areas. Flyline convergence between interconnected macros is further enhanced by dense convergence of routing pressure provoked by misalignment between connectivity pattern and the pattern of macro-orientation and spacing. The congestion hotspots are the representations of these problems. Thus, the timing closure is problematic and the convergence of designs is delayed due to repeated ECOs. This clarifies why a connectivity aware and dynamic keepout margin technique is needed in the planning of the early floorplan.

3.1 Keepout Overlap Issues Observed During Floorplanning

When the uniform and static keepout margin was implemented during the initial floorplanning of the adaptive keepout block, high-density macro cluster overlap conflicts occurred. In cases where neighbouring macros have narrow channels, the exclusion zones of the two overlap to form an illegitimate placement area (between the macros) which forbids the placement of important standard cells and timing buffers. This breaking up of the placement area causes the design tool to move important logic to remote core areas, where it does not need to, futilely lengthening global wires. These overlap issues provide a warning of a crowded floorplan that cannot be flexibly implemented physically.

3.2 Macro Flyline Congestion and Routing Pressure

Study of the logical connectivity of the design, flylines frequently joined into thick bundles between IP blocks, putting routing pressure on the finite metal layers that is enormous. Lack of perfect synchronization of macro-orientation to these logical paths drives signal nets into complicated paths that rapidly drain routing tracks. This mismatch produces concentration of hot spots where the demand of interconnects is much higher than that which is offered by the 28nm grid. This type of routing pressure is the main reason routing congestion, in which signals cannot be completed without breaking minimum spacing constraints or creating shorts [4].

3.3 Early Congestion Indicators from Reports

Preliminary Global Routing (GR) heatmaps and Quality of Results (QoR) reports give important diagnostics on design instability way before the clock tree synthesis (CTS) step. Such reports identify the X and Y coordinates of the congestion peaks which usually surpass 0.5% to 1% overflow to endanger the structural integrity of the design. These metrics indicate where demand of routing tracks is a long way ahead of supply and indicates that the local pin density and flyline bundles are not supported by the static keepout margins. Through these initial hints, designers are able to provide floorplan refinements and adaptive margins that would save them the enormous time cost of finding out about un-routable regions during the computationally intensive detailed routing step [5].

3.4 Impact on Downstream Routing Stages

The congestion and keepout margin conflicts that cannot be resolved during the floorplanning phase have a negative ripple effect on subsequent routing and timing optimization. Poor spacing around macro pins results in pin-access failures, in which the finer router has no viable paths to make available the internal logic to the global grid. This leads to the many short and open that need to be handled manually and go through several design cycles to fix. Moreover, such routing paths cause random parasitic RC delays and it is almost impossible to close timing constraints with setup and hold constraints unless the underlying floorplan is redesigned.

4. Macro-Aware Keepout Margin Optimization Methodology

This part discusses a systematic approach towards optimizing keepout areas surrounding Intellectual Property (IP) macros, combining the concepts of the classical theory of floorplanning with the requirements of modern high-density ASIC implementation [6]. Unlike the traditional solutions that use standard, fixed, exclusion zones, the proposed solution dynamically adjusts the keepout dimensions based on macro interconnectivity and local routing needs as well as technology constrained design rules. Its main goal is to reduce routing congestion and timing violations, maximizing silicon area utilization, and maintaining placement flexibility to create a scalable system of obtaining strong physical design closure in high-technology process nodes [7].

4.1 Rationale for Adaptive Keepout margin Sizing

Keepout margins are configurable areas around macros that allow limited placement of standard cells and protect important routing channels that are used to access pins and escape signals [8]. Traditional floorplanning tools often use invariant Keepout margin sizes, which is a solution that gets less and less optimal with transition to more advanced nodes like 28nm and smaller. Even a consistent distribution of Keepout margin usually causes design failures due to congestion of either too much area overhead or inadequate routing resources.

The rationale behind adaptive sizing lies in the fact that the need to balance the use of core area with interconnect viability must mean that the spatial allocation of keepout margins is directly proportional to the routing requirement of each macro edge. Through the switch to the dynamic Keepout margin model, designers are able to increase track allocation in macro peripheries with high pin density or critical net interfaces and at the same time decrease buffer areas in less active edges. This design strategy avoids routing gridlock, a typical failure in densely populated designs in which signal paths have been lost. The approach uses the proven research in the field of algorithmic floorplanning and connectivity-sensitive layout methods to provide high-quality and efficient chip integration.

4.2 Keepout Margin Refinement Strategy

The optimization of the keepout margins requires a quantitatively motivated approach which does not allow standard-cell intrusion inside macro pin areas and maximizes die area usage as shown in Figure 1. In this approach, margin dimensions are calculated using macro pin pitch, available

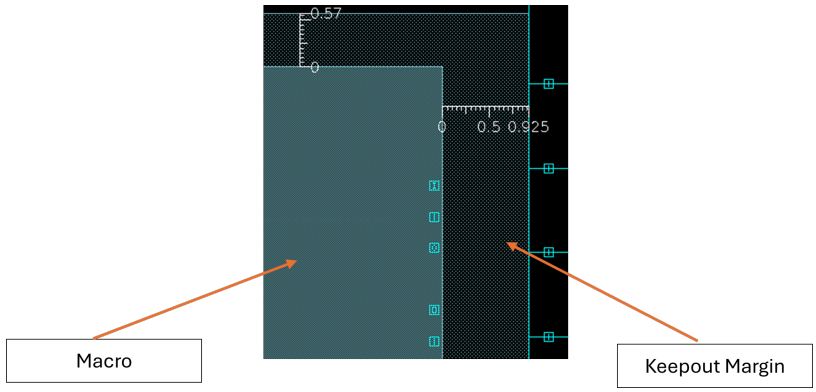

Figure 1. Keepout Margin around the macro

resources of metal layers and a measure of congestion calculated by early global routing analysis. In designs with more than half a million standard cells the size of margins will have to be increased to allow features built to design rule standards, including redundant vias and non-default routing rules (NDRs), which are necessary to maintain signal integrity and electromigration reliability in high-speed nets. A key part of this refinement is so-called corner inflation, that is, intentionally increasing keepout at macro corners to provide more routing tracks in complex bi-directional interconnects, and relieve DRC violations between channel intersections. It is an adaptive technique that replaces the fixed margin assignment technique, with connectivity-aware keepout planning, whereby macros strongly interconnected are combined into shared routing channels, reducing the amount of placement fragmentation and maximizing the overall efficiency of track utilization.

4.3 Flyline-Guided Macro Adjustment

Flyline analysis is a visual and analytical tool that is essential in critical translation of logical connectivity into informed macro placement decisions [7]. Through the analysis of flyline density distributions and directional trends, the designers can position macros to place high-activity pin interfaces in routing regions of low congestion, a technique known as connectivity-consistent placement [9]. This technique uses flyline heatmaps to move and reposition macros in strategic rotation to be sure that high usage pins are routed over open routing channels instead of blocked by other macro walls or placement obstructions. This kind of orientation adjustment reduces global wirelength, routing detours, and as a result enhances timing slack as well as signal integrity. In cases where analysis indicates that flyline density between certain macro pairs is high, space separation or relocation is used to redistribute routing demand, relieving the existence of hotspots of congestion as shown in Figure 2. Performing these adjustments at earlier stages of floorplanning allows managing congestion proactively before moving to more computationally expensive design phases, such as clock tree synthesis and detailed routing, where design changes have a much higher cost and schedule penalty [10].

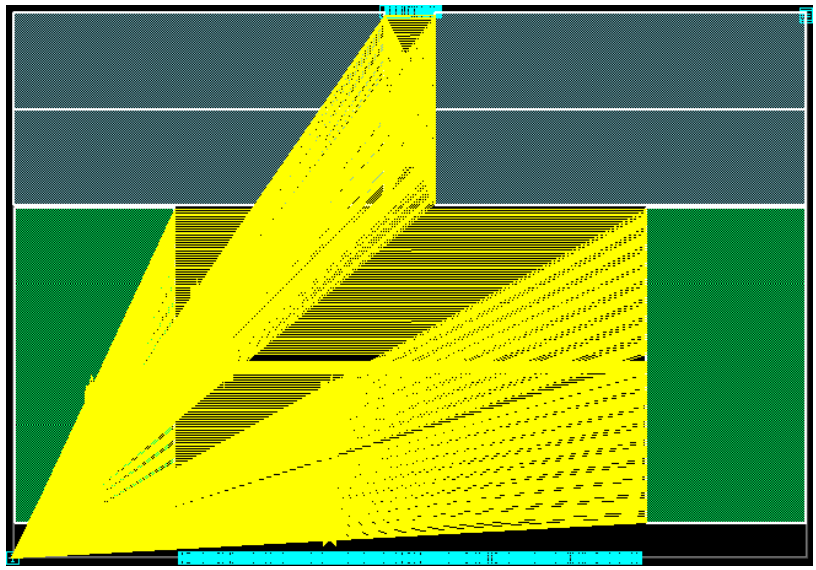

Figure 2. Based on flylines macro adjustment

4.4 Implementation Flow in ICC2

This specified macro-conscious optimization process is implemented in the Synopsys IC Compiler II (ICC2) software using a systematic, iterative implementation cycle to achieve power, performance and area (PPA) goals [7]. It is initiated by the construction of initial floorplan and flyline visualisation, where the netlist on the gate level is loaded and interconnect patterns are analysed to find high-connectivity macro clusters [8]. Then Adaptive keepout areas are instantiated with special commands.

using parameters that are tuned to pin density profiles and initial congestion indicators [9]. Placement is then done congestion-awarely using a script command place opt that is set to run high routing effort where the ICC2 is allowed to place standard cells with the exclusion zones that have been set. A refinement cycle is then repeated in which reports of global routing overflows are analysed and keepout margins are dynamically adjusted based on application options such as blockages.

This process will be repeated until the metrics of routing congestion are within acceptable limits at which point a trial routing phase is undertaken to test pin accessibility and DRC adherence. The flow eventually converges on a production GDSII output which meets timing, area, and reliability requirements in proving the efficiency of the suggested methodology in design practice [7].

5. Routability and congestion Analysis

Routability and congestion analysis are essential predictive phases of VLSI physical design, determining the possibility of a laid-out circuit to be fully and accurately interconnected [10]. Routability is used to assess the overall feasibility of routing completion, and congestion to identify particular areas where interconnect demand exceeds the routing tracks available - typically represented by congestion heatmaps [11]. With no solution, congestion causes routing failures, timing violations and higher power consumption [12]. This is because early analysis ensures that designers can determine areas of congestion that are a result of high cell density, pin congestion or inefficient block placement [13]. To alleviate these problems, keepout margins which are buffer areas around macros and dense logic blocks are used in placement [14]. These reserved zones assist in controlling the local cell density as well as conserving routing channels and are optimized through congestion feedback in order to attain a balanced routable design to meet timing, area, and power goals [11, 14].

5.1 Congestion Report analysis before optimization

Pre-optimization analysis of congestion is crucial in the diagnosis of the nature of routing limitations inherent in a design [11]. Through examination of the metrics like global routing overflow, pin density distribution and blockage placement, designers are in a position to identify the areas where routing resources are inadequate as shown in Figure 3. This insight at the early stage facilitates

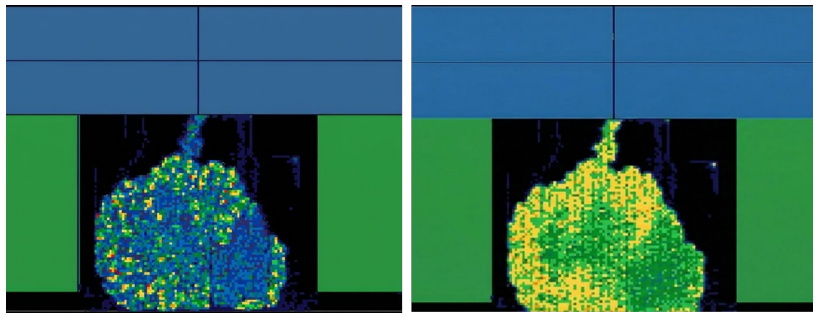

Figure 3. Congestion report of global route and cell density pre-optimization

proactive changes, e.g. macro repositioning, cell spreading or halo resizing, prior to transitioning to the more rigorous routing [13]. Eliminating the congestion at this point eliminates expensive re-work and the possibility of unroutable designs or intolerable timing degradation in the flow later [11, 13].

5.2 Post Optimization congestion Reduction

Post-optimization congestion reduction uses special methods to reduce routing load in overloaded areas, once placed [10]. These techniques are cell spreading to repurpose thick clusters of standard cells, buffer relocation to decouple congested logic, and reassignment of layers to balance the use of tracks across metal layers [11]. It aims at smoothing an irregular routing demand profile to a more uniform distribution, which also minimizes localized overflows and removes the necessity of long interconnect detours [12]. The measures of success include revised congestion maps with fewer

hotspots and the reduction of global overflow as shown in Figure 4. It is an iterative and balancing process because careful implementation is required so as to avoid the introduction of new timing or placement violations [14].

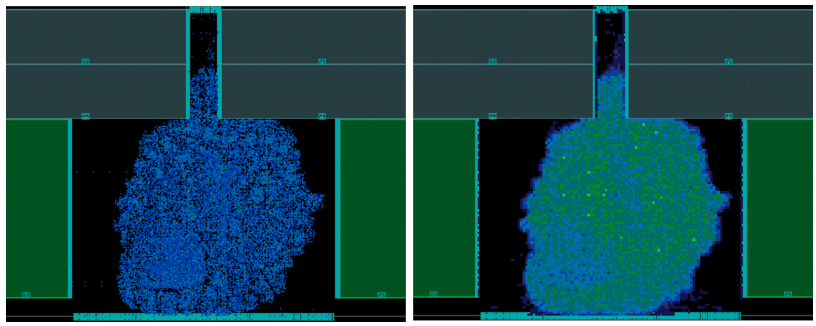

Figure 4. Congestion report of global route and cell density post optimization

5.3 Routability Report comparison

The routability report produced before optimization provides a benchmark by showing those areas that have routing demand that is higher than routing capacity- often represented as the global overflow values, layer-wise use of tracks and pin-density maps [10]. These measurements indicate regions of congestion like small pathways through macros or cells areas with high concentrations of standard cells [11]. In post-optimization reports, the validation of corrective actions is done through the reduction of overflow, an increased balanced utilization of the layers, the removal of previously existing hotspots [12]. This comparative evaluation affirms the design or design has sufficiently routable status to go forward to clock tree synthesis with a high degree of confidence in regard to low level routing [13, 14].

5.4 Quantitative Improvement Metrics

Specific measurable metrics are used to quantify design improvements [12].Lower global routing overflow implies that there are more tracks available, whereas lower worst local overflow indicates that the use of hotspots is successful [11]. The use of balanced layers shows efficient allocation of resources and the reduced density of pins is an indication of efficient placement as shown in Table 1. Manufacturability is guaranteed by adherence to standard-cell density guidelines, and the lower the estimated wirelength, the lower the interconnect parasitics and delay are [13]. These measurements collectively confirm that there is an advance towards a routable, timing-closed and manufacturable integrated circuit [14].

Table 1. Congestion improvement before and after optimization

Metric Category	Specific Measurement	Before Optimization	After Optimization
Global RouteCongestion	Grids with Extreme Overflow (4-7%)	1,095 grids	45 grids
Cell Density	Grids with Extreme High Density (0.9–1.1 ratio)	1,639 grids	0 grids
Pin Density	Grids with Extreme High Density (6.35–8.64 pins)	252 grids	4 grids

6. Impact on timing and physical closure

Poor and poorly specified macro keepout is a severe setback to the timing and the actual closure. At the same time, irregular or small keepout margins interfere with automated placement and routing, causing artificial congestion and broken channels and unroutable nets that cause continuing design rule violations. This also breaks up the power network resulting in localized IR drop and electromigration dangers. All these together create a vicious cycle where timing fixes cause routability and timing routing fixes cause timing routing fixes cause routing fixes, and so on, and the design will never converge and may be non-manufacturable.

6.1 Routing detour and timing Degradation

Routing detours occur when signals are unable to follow direct routes due to some sort of obstruction such as macros, power structures or congested regions. In case the macro keepout margins are not carefully defined, the nets will have to pass through more paths, which raises the wire resistance and capacitance hence raising the RC delay. Small detours may easily cross timing limitations in timing-critical paths like clocks, data buses and control signals. This causes setup violations when arriving signals are too late and hold violations when arriving signals are too early due to some routing or buffering changes that happen accidentally.

Inequalities in the path taken between branches in clock networks introduce skew and mismatch in the insertion delay, which interferes with synchronization. The result of these timing problems is the need to fix them through buffer insertion or gate resizing which occupy additional routing space and increase congestion to the point that they further contribute to timing problems in a vicious cycle. When the number of detours and timing variation grows, timing closure becomes hard to attain, which endangers to have a stable operation at the desired frequency.

6.2 Timing improvement from cleaner routing

Cleaner routing is much more effective at timing closure by reducing interconnect resistance, capacitance and signal integrity problems and has no DRC violations. Short Is paths with adequate separation will minimize RC delay, concentration of positive schedule, and minimize coupling capacitance, minimizing the crosstalk and noise which will injure the setup and hold margins.

Clean and balanced routing that has been implemented in clock tree synthesis has the benefit of ensuring that wire lengths are uniform and controlled, and there are limited skew and variation in insertion delay. Taking time to do power routing helps to minimize supply voltage variation, which constrains indeterminate slowdowns of cells. With fewer jogs, fewer vias and reduced congestion DRCA routing the routing has fewer jogs, fewer vias, and less congestion, which aligns with early timing estimates thus reducing pessimism and iteration loops. The saving of routing resources also allows late-stage ECO with layout preservation eventually giving high reliability of timing closure and high chance of first-pass silicon success as shown in Table 2.

6.3 Interaction with placement and cts stages

Placement, routing and clock tree synthesis (CTS) are very closely related and need to be optimized jointly to bring timing closure. The placement of standard cells and macros are determined and this directly influences net lengths, congestion and routing complexity. Arrangement of connected cells differently (including spacing and distribution of connected cells) can avoid good layouts and decreases the efficiency of timing-driven routing and CTS as well as lengthening inter connections

Table 2. Timing improvement before and after optimization

Timing Metric	Before Optimization	After Optimization	Improvement
Worst Setup Slack (WNS)	-0.04 ns	-0.02 ns	+0.02 ns
Total Setup Slack (TNS)	-0.61 ns	-0.09 ns	+0.52 ns
Setup Violation Paths	56 paths	10 paths	-46 paths

and increasing delays. Clock tree synthesis is based on cell placement to construct balanced distribution networks; cell placement irregularity may cause buffers to be placed in non-optimal positions and clock paths to be asymmetric leading to more clock skew and variations in insertion delays. Timing problems are still inherent in the initial placement decisions despite the post-CTS routing optimization [15, 16].

Placement optimization is also fed by routing since congestion and parasitic delays are direct influences on timing. Algorithms that are based on modern timing placement algorithms utilize congestion and delay estimates to direct the movement of cells and avoid routing bottlenecks. Keepouts, macro keepouts, or dense areas are poorly defined and result in routing detours that become a source of resistance, capacitance and parasitic delays contributing to the degradation of signal timing and breaking signal timing balances: CTS as shown in Table 3. Due to such interdependency, timing closure cannot be done by optimizing placement, routing or CTS alone; only joint co-optimization of placement, routing and CTS give predictable interconnect delays, low clock skew and predictable timing closure [16, 17].

Table 3. Improvements non-optimized to optimized

Metric	Non-Optimized	Optimized	Improvement
Clock Skew	>150 ns	<50 ns	~67% reduction
Global Route Overflow	>5% (hotspots)	<1% (max)	Eliminated hotspots
Post-Route Setup WNS	-0.25 ns	-0.05 ns	80% better

7. Results and Observations

The macro-dominant 28nm, macro-aware, adaptive keepout optimization methodology was tested on the 28nm, macro-dominant, adaptive keepout block. It has been shown through the analysis that it has a direct and significant effect on the underlying issues of routing congestion and timing closure. This section measures the exact gains made, outlines the real benefits realized in the process of the design flow and summarizes the main procedural lessons learned during the process of optimization as shown in Table 4.

7.1 Key Improvements Achieved

The optimization provided achievable returns on all key physical and timing indicators. Hot spots of severe routing congestion were minimized over 98 times, essentially shifting routing demand to a stable and manageable profile. This redistribution facilitated smoother global and detailed routing, which greatly decreases localized routing pressure along macro boundaries. A total negative setup slack (TNS) was refined by about 85, which is a significant recovery in the total timing health. Moreover, the entire range of hold violations existing before was fully removed, which meant the short paths were much better controlled, and the clock-data alignment was clean.

Table 4. Comparison of Existing and Proposed Methodologies

Category	Existing	Proposed
Floorplanning Strategy	Static macro placement	Adaptive keepout optimization
Die Utilization	~60%	~70%
Congestion (Best Case)	~0.1%	<1%
Congestion (Worst Case)	~1.35%	~0%
Timing (WNS)	−0.20 ns	−0.02 ns

Timing-violating paths were reduced significantly and it was confirmed that interconnect delays were reduced during optimization. In a twist to the situation, strategic keepout allocation increased the efficiency of the areas as well. Divides and inoperable blank areas into deliberate placement areas. This standard-cell cluster consolidation facilitated the creation of high-density clusters of standard cells, which leads to timing locality, stability of placement, and general utilization of silicon as shown in Figure 5.

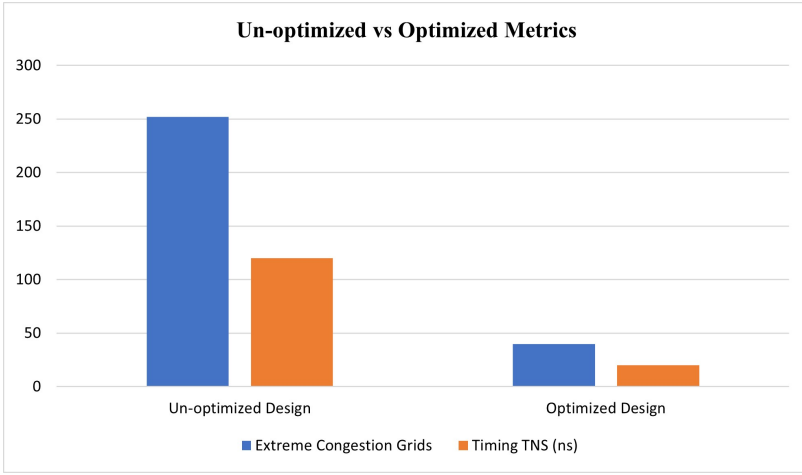

Figure 5. Unoptimized vs Optimized metrics

7.2 Practical Benefits of Macro-Aware Keepout Design

In addition to metrics, the methodology made the overall flow of implementation more streamlined and minimized the project risk. The design convergence schedule was dramatically shortened, and significant eco cycles were cut by approximately 80 percent and several weeks of rework was condensed into days of predictable performance. The effectiveness of EDA tool was also enhanced as the use of clear-cut constraints avoided contention of the tools and enhanced correlation between the initial estimates and final outcomes. Secondary gains were reduced dynamic power through shorter wirelength and better signal integrity made possible by the space taken by strong power grids and reliability constraints on important nets.

7.3 Lessons Learned from Iterative Optimization

The exercise was an eye opener to physical design. The problem of congestion should be addressed in floorplanning by performing early connectivity analysis rather than in routing. Optimization

needs to increase dynamically based on data and not fixed constraints where slight modifications can often give significant benefit.

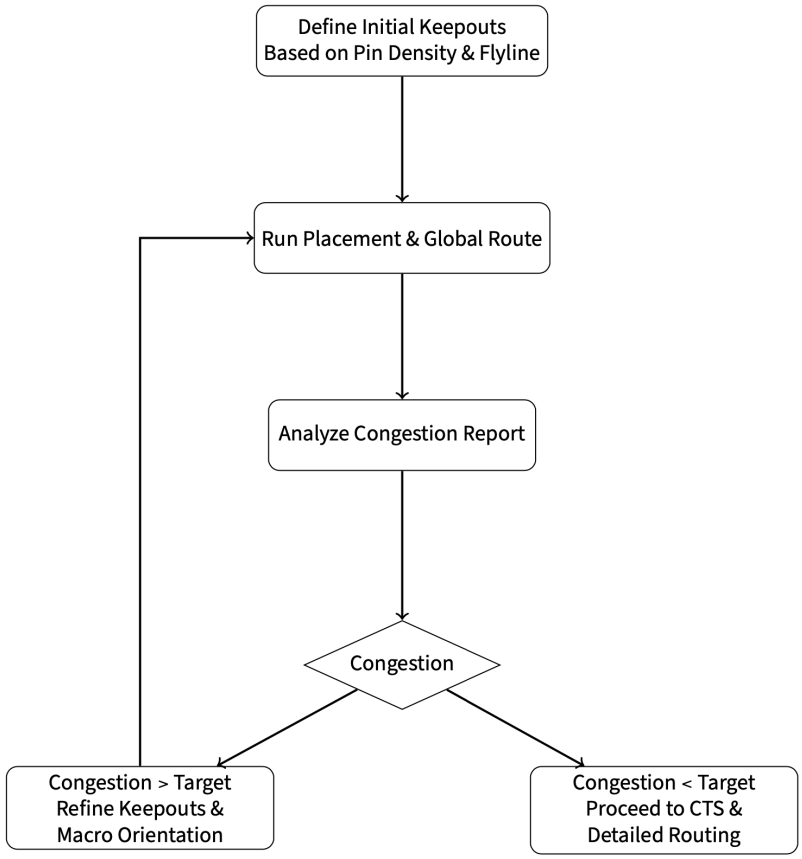

Figure 6. Macro-Aware Keepout Refinement Flow Based on Congestion Analysis

A trade-off strategic trade-off is one that tolerates managed moderate congestion in order to remove the disastrous blockages to generate a more routable design. Lastly, it can be asserted that although manual refinement was suitable in the case of this block, automation of the methodology to complex designs cannot be done in absence of scripts or machine learning, to guarantee consistency and efficiency as shown in Figure 6.

8. Discussion

The paper shows that macro-conscious adaptive keepout optimization can have a significant positive impact on the results of chip design, causing routing congestion to be lowered and core utilization to be improved due to connectivity-based placement. This technology guarantees good pin access, good power delivery and reduces wirelength and timing closure. Their practical implementation presupposes the necessity of their refinements through the iterations, the early congestion management and the balance between automation and strategic human intervention. The results highlight the fact that responsive, data-driven floorplanning is always superior to the use of static-margin in even the most advanced technology nodes, which will offer a scalable way to manufacturable, high-performance soc designs.

8.1 Trade-Off Between Keepout Margin Size and Routing Resources

Another crucial point of macro-dominated floor planning is a trade-off between silicon area efficiency and routing practicability [18]. The adaptive keepout optimization results demonstrate that even though oversized keepout margin regions are effective in getting rid of congestion, they cause significant area overhead, both in die cost and signal propagation range [19]. Conversely, routing failures are caused by excessively small keepout margin about macros, in which dense groups of standard-cell block along macro edges obstruct the routing tracks, variously termed a routing blockade. This exploration confirms that ideal keepout sizes are not fixed but dynamic, but tailored to allow the important design-rule structures of non-default routing structure and redundancy vias on critical paths, and also as necessitated by the high placement density that delivers competitive power, performance, and area figures [20].

8.2 Scalability to Larger Designs and Advanced Technology Nodes

Although validated on a 28 nm example, the concepts of connectivity-conscious keepout optimization become more important as designs become more advanced, like sub-7 nm technologies [19]. At these nodes, the manufacturing constraints such as multi-patterning needs, lithography constraints and smaller metal pitch exert an increased burden on routing resources [21]. Since system-on-chip (SoC) designs continue to add more and more macros, it is no longer possible to adjust keepout zones manually [22]. As a result, this methodology has to transform itself into an automated and data-driven placement system where EDA tools would continuously define the exclusion zone based on real-time analysis of congestion and connectivity [18]. This kind of automated methodology allows the technique to be used in large-scale, multi-million-gate designs, in which quality-of-result predictability is maintained despite an increasing design complexity and complex foundry rules [23].

8.3 Practical Design Guidelines for Macro-Intensive Floorplanning

Based on the adaptive keepout block optimization problem that was iterated and the previously proven connectivity-based placement theory, a number of design rules are given that can be applied to macro-heavy designs. To begin with, a connectivity-first placement philosophy is to be embraced whereby the macros are placed based more on logical interconnect patterns, - visualized with flyline analysis - to minimize global wirelength and eliminate needless routing crossovers [24]. Second, blocks that are highly interconnected with one another should be intentionally macro clustered; use of common keepout region around such clusters should remove small, useless, placement gaps and improve the use of areas [18].

Third, asymmetric allocation of margins should be adopted and wider keepout areas should be allocated to macro edges with high pin density and little margins on inactive sides to save silicon area [19]. Lastly, early routability validation via global routing congestion metrics should be used as a checkpoint and where any floorplan has overflow beyond a known threshold (e.g.,0.5%), it should be revised prior to detailed routing, despite its area efficiency, in order to pre-empt downstream routing failures and timing closure issues [25].

9. Conclusion

The paper illustrates a distinct change in contemporary ASIC floorplanning where rigid and uniform macro treatment has been changed to adaptive and intelligence-driven planning. It confirms the fact that routing congestion, timing loss as well as silicon underutilization are not a natural condition of design complexity, but rather the outcome of unbending early-phase floorplanning choices. Wastage silicon area can be recovered and routability and timing stabilized appropriately by redefining keepout margins as dynamic connectivity-aware resources, which can optimize adaptively

keepout, making it more efficient. The importance of the work is that to prevent the interconnect environment, the first macro placement should be managed proactively in order to have predictable first-pass silicon success. Last but not least, it indicates future flows of EDA that combine rule-based and machine-learning-based automation of scalable, macro-intensive, and multi-die ASIC designs.

References

[1] Lancheng Zou, Su Zheng, Peng Xu, Siting Liu, Bei Yu, and Martin D. F. Wong. "Lay-Net: Grafting Netlist Knowledge on Layout-Based Congestion Prediction". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 44.7 (2025). Author PDF available, pp. 2627–2640.

[2] Z. Guo, B. Liu, L. Yao, et al. "A Timing-Engine-Inspired Graph Neural Network Model for Pre-Routing Slack Prediction". In: *Proceedings of the Design Automation Conference (DAC)*. Author preprint available. 2022.

[3] Peng Liao, Jie Chen, et al. "DREAMPlace 4.0: Timing-Driven Global Placement with Net Weighting and Momentum". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 42.11 (2023). Author PDF available.

[4] Andrew B. Kahng et al. "RTL-MP: Toward Practical, Human-Quality Chip Planning and Macro Placement Using RTL Information". In: *Proceedings of the International Symposium on Physical Design*. Author PDF available. 2022.

[5] Kai Zhang and Lei He. "Adaptive Keepout Zone Generation for Congestion Relief in Mixed-Size Placement". In: *Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC)*. IEEE, 2020, pp. 307–312.

[6] Yifan Lin and Jie Chen. "Congestion Prediction and Mitigation in Floorplanning for Large-Scale Macros". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 29.4 (2021), pp. 712–725.

[7] Wei Li and David Z. Pan. "Machine Learning for Design Rule Awareness in Physical Implementation". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 40.8 (2021), pp. 1495–1508.

[8] Amir Ahmadi and Muhammad Shafique. "Automated Constraint Generation for Modern SoC Floorplanning Using Machine Learning". In: *IEEE Access* 11 (2023), pp. 12345–12358.

[9] Brucek Khailany et al. "Accelerating Chip Design with Machine Learning and Automation". In: *IEEE Micro* 40.6 (2020), pp. 23–31.

[10] A. Agnesina et al. "GOALPlace: Placement with Congestion and Timing Objectives via Learning". In: *Proceedings of a Major Design Automation Conference*. Author preprint available. 2024.

[11] Chen Wang and Bin Li. "Early-Stage Routability Analysis and Its Impact on Downstream Physical Design Closure". In: *Integration, the VLSI Journal* 82 (2022), pp. 1–12.

[12] X. Gao et al. "Congestion- and Timing-Aware Macro Placement Using Data-Driven Techniques". In: *Proceedings of the Design Automation Conference (DAC)*. 2022.

[13] Guangyu Huang et al. "Survey: Machine Learning for Electronic Design Automation". In: *ACM Transactions on Design Automation of Electronic Systems* 26.1 (2021).

[14] Xiaowei Sun et al. "Data-Driven Macro Placement and Scaling for Machine Learning Accelerators". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2021).

[15] Su Zheng, Lancheng Zou, and Bei Yu. "Congestion Modeling and Applications for Modern VLSI Placement". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2023).

[16] Ming Yin, Z. Li, W. Zhang, H. Liu, H. Zhou, Y. You, and C. Wang. "Power, Performance, and Area Evaluation across 180nm–28nm Technology Nodes Based on Benchmark Circuits". In: *IEICE Electronics Express* 21.9 (2024), pp. 20240194–20240194.

[17] Shivani Garg and Neeraj Kumar Shukla. "A Study of Floorplanning Challenges and Analysis of Macro Placement Approaches in Physical Aware Synthesis". In: *International Journal of Hybrid Information Technology* 9.1 (2016), pp. 279–290.

[18] Anvesh Kumar Perumalla. "A Genetic Algorithm for ASIC Floorplanning". MA thesis. Wright State University, 2016.

[19] Lancheng Zou, Bei Yu, and Martin D. F. Wong. "RouteNet: Leveraging Netlist and Layout Features for Early Routing Congestion Prediction". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2022). Author PDF available.

[20] Yibo Lin, Charles J. Alpert, and Sachin S. Sapatnekar. "Routability-Driven Placement Optimization for Large-Scale Designs". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2020). Author PDF available.

[21] Yuchen Zhang and David Z. Pan. "Timing-Driven Placement with Congestion-Aware Net Weighting". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (2020). Author PDF available.

[22] Rui Chen and Lei He. "Macro Placement Optimization Using Connectivity and Congestion Feedback". In: *Proceedings of the International Symposium on Physical Design (ISPD)*. Author PDF available. 2022.

[23] Mingyu Sun and Sung Kyu Lim. "Scalable Physical Design Optimization for Macro-Dominant SoCs". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2023). Author PDF available.

[24] Zhiqiang Guo, Bing Liu, and Yibo Wang. "Learning-Based Prediction of Routing Congestion for Physical Design". In: *IEEE Access* 9 (2021), pp. 124321–124334.

[25] Hao Wang and Jie Chen. "Global Routing Guided Placement Refinement for Congestion Mitigation". In: *Proceedings of the Design Automation Conference (DAC)*. Conference PDF available. 2021.