Net Weighting to Reduce Repeater Counts during Placement

Brent Goplen*
Department of ECE
University of Minnesota
Minneapolis, MN 55455
bgoplen@ece.umn.edu

Prashant Saxena*
Advanced Technology Group
Synopsys Inc.
Hillsboro, OR 97124
saxena@synopsys.com

Sachin Sapatnekar
Department of ECE
University of Minnesota
Minneapolis, MN 55455
sachin@ece.umn.edu

ABSTRACT

We demonstrate how to use placement to ameliorate the predicted repeater explosion problem caused by poor interconnect scaling. We achieve repeater count reduction by dynamically modifying net weights in a context-sensitive manner during global placement and coarse legalization. Our scheme, which models layer assignment as well as valid inter-repeater distance ranges, can decrease the repeater counts significantly with minimal impact on wirelength.

Categories and Subject Descriptors

B.7.2 [**Integrated Circuits**]: Design Aids – *Placement and routing*; B.7.1 [**Integrated Circuits**]: Types and Design Styles – *Advanced technologies*

General Terms

Algorithms, Performance, Design, Experimentation

Keywords

Placement, Net weighting, Force-directed placement, Repeater, Buffering, Scaling, Interconnect.

1. INTRODUCTION

With decreasing feature sizes, interconnect delays do not scale as well as gate delays. Consequently, they are becoming a dominant part of the total delay in deep submicron technologies, especially as overall chip areas do not shrink with scaling. Since the delay of an unbuffered wire grows quadratically with wirelength, repeaters are needed to linearize this delay, as well as to restore signal slews. However, the inter-repeater separation unfortunately scales poorly (0.586x per generation, in contrast to the normal shrink factor of 0.7x) [1], leading to an explosion in the expected number of repeaters and a consequent breakdown of many of today's CAD algorithms and methodologies [2]. As repeaters become more prominent in future designs, they are predicted to cause a number of problems including increased power consumption and degraded design convergence.

* This work was done while B. Goplen and P. Saxena were with Intel Labs (CAD Research), Hillsboro, OR.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
DAC 2005, June 13–17, 2005, Anaheim, California, USA.
Copyright 2005 ACM 1-59593-058-2/05/0006…$5.00.

As the number of repeaters increases, it becomes more difficult to integrate them into the design. Efforts are needed to keep their number minimized while satisfying traditional design constraints such as performance, power and area. It has been observed that the signal propagation speed on a long buffered wire does not vary appreciably over a significant range of inter-repeater distances [3]. This allows the inter-repeater distances to be increased slightly without compromising the performance significantly, thus allowing fewer repeaters to suffice. However, although this method can help reduce the number of repeaters on individual nets in a *placed* design (at the cost of slightly degraded delays and slews), it does not modify the placement taking the global view of the entire netlist into consideration. In this paper, we show that it is possible to reduce the overall repeater count even further during placement by trading off the lengths of nets that are just on the threshold of requiring (additional) repeaters, against those of less-critical nets that can afford to grow in length without needing more repeaters. This can be accomplished by the judicious application of context-sensitive net-weights to these nets. However, if it is to be practical, any proposed net-weight modification scheme must rely on only a small number (ideally, one) of parameters. Furthermore, it should be robust enough that its controlling parameter(s) requires no individual tuning for each net, or even for each testcase. Another complication in any such scheme is introduced by the fact that the number of repeaters required on a net is dependent on the layer assignment and routing of that net – information that is unknown at the time of the placement because of the unaffordable cost of invoking global routing within each placement iteration.

Net weighting has previously been used primarily in the context of timing-driven placement and low-power design, in order to reduce the lengths (and consequently, wire loads) of critical nets [4] [5] [6] and reduce the power consumption [7] [8] [9]. We use net weighting in a completely novel way, *viz*., to nudge nets away from repeater insertion and towards deletion thresholds. In this paper, we modify industrial implementations of the Kraftwerk global placer [10] that incorporates native repeater modeling [11], as well as the force-directed-Mongrel (FD-Mongrel) coarse legalizer [12] [13], in order to investigate the effects of net weighting on repeater count reduction during placement.

Being essentially a net weight modification scheme, our method is quite general in scope – as can be seen from its successful application to two different placement algorithms (*viz*. KraftWerk and FD-Mongrel) that is described in this paper. In general, any placer that supports net weights can benefit from this scheme; this includes the vast majority of commercial and academic placers. The empirical layer prediction and repeater prediction models that are used in our experiments are orthogonal to the choice of the specific placer used.

2. PLACEMENT INFRASTRUCTURE

At the global placement stage, we apply our net-weighting scheme within the force-directed placement paradigm, using an implementation of Kraftwerk. Kraftwerk extends the traditional quadratic analytical model (in which one seeks to minimize the weighted sum of the squared Euclidean distances of connected cells, in analogy with finding the equilibrium for a system of springs) by introducing a spreading force field. The forces in this spreading field are computed using the density profile of the cells in the design. Furthermore, our implementation leverages the "linearization" of the quadratic objective function [14] that usually results in improved solution quality.

This implementation of Kraftwerk has been augmented further with MorePlace [11], which is a scheme to model repeaters natively during analytical placement. This scheme allows the system to avoid the massive perturbations caused when the large numbers of repeaters required at future process technologies are patched directly into the netlist in an interleaved or iterated fashion during global placement [11]. In MorePlace, virtual repeaters are added and deleted as needed during each iteration of Kraftwerk, contributing repulsive and/or attractive forces (in addition to the usual density-derived forces for spreading) without fragmenting the original netlist. Not only do virtual repeaters contribute to the spreading forces but also provide attractive nets with quadratic costs that cause them to spread out equidistantly along their nets. The virtual repeater insertion and deletion is done at the beginning of each Kraftwerk iteration, prior to the calculation of the new cell positions (including the virtual repeaters), using a length-based repeater prediction scheme [15] [11]. If the average inter-repeater distance along a net is below a minimum value, repeaters are deleted from the net. On the other hand, if the average inter-repeater distance is greater than a maximum value, one or more repeaters are added to the net. The difference between these maximum and minimum values captures the range of tolerable inter-repeater distances [3]. Finally, the surviving virtual repeaters are instantiated after the placement terminates; the repeater force model helps ensure that sufficient space is available to do so.

After global placement with Kraftwerk/MorePlace, we apply our net weighting scheme to FD-Mongrel [13] that is used for coarse-grained legalization. FD-Mongrel uses a hybrid approach that maintains the quality of the force-directed placement while making significant improvements on overlap removal. It begins with a coarse grid approach that uses forces to remove overlaps globally. In the second phase, detailed placement is performed using a fine grid in which cells are ripple-moved from the densest bin to the least dense bin following a monotonic path of least resistance, i.e., the path causing the least amount of quality degradation. Ripple-moves that result in wirelength constraint violations are avoided. The cost (or gain) computed for each potential move using the wirelength and net constraints is used to determine which cells to move during the detailed placement phase of FD-Mongrel. Finally, the almost-legal placement from the coarse legalization is passed onto a fine-grained legalizer.

3. PROPOSED APPROACH

In our approach, net weighting is used at the global placement and coarse legalization stages to reduce the number of repeaters needed in the placement. Assume, for the purpose of illustration, that we introduce a repeater on a wire every x microns because of signal slew constraints. Then, if we have two approximately equi-critical wires of lengths $1.05x$ and $1.5x$ respectively, a simple length-based repeater insertion engine will add a repeater to each of them. However, if we shrink the length of the first one marginally to $0.95x$, even if it means allowing the other net to grow to a length of $1.6x$, we can avoid one of the repeaters without violating our slew constraints, having a significant impact on performance or degrading total wirelength. This is the intuition behind our approach.

However, in practice, the decision of when a repeater should be inserted is considerably more complicated. Not only are ranges of inter-repeater distances acceptable (in contrast to the sharp threshold of x in our simple illustration), inter-repeater distances also depend intrinsically on the layer on which the net will eventually be routed. At the same time, one cannot afford to invoke a global router within each placement iteration because of runtime constraints. This leads to the need for a mechanism to predict the repeater needs of a net. Fortunately, this problem is not as intractable as it seems. Many industrial flows not only use length-based schemes[1] for repeater insertion (on all but a few high-fanout critical nets) [15], they also use length-based schemes to decide the layer assignments for different nets. The former heuristic owes its existence to the fact that most nets in a mapped design are two-pin nets, and greedy length-based repeater insertion on such nets is almost as good as more sophisticated dynamic programming based approaches, with the advantage of being much faster. The correlation between net length and layer assignment arises from the observation that short wires are best routed on the more resistive lower layers, while longer wires benefit more from the upper layers where their improved wire delays amortize the via stack penalties. Furthermore, since routing architectures usually follow preferred direction routing on each layer, process designers often architect pairs of adjacent metal layers to be electrically similar. Thus, metals M3 and M4 may form a pair, as may metals M5 and M6 (the upper metal layers are usually not available for block level synthesis).

Consequently, we use the length-based repeater prediction scheme described in [11], which has been validated against tape-out data. In this scheme, l_{rep}^{M} is the optimal inter-repeater distance on metal layer M and can easily be determined using simulation as in [2]. We insert a new repeater on a net routed on M only if its length is greater than $1.4\,l_{rep}^{M}$; similarly, we delete an existing repeater when the inter-repeater distance has shrunk to less than $0.7\,l_{rep}^{M}$. We assume that since short nets will be routed on the lower metal layers, we can use l_{rep}^{M3} (or the average of l_{rep}^{M3} and l_{rep}^{M4}) to determine their repeater needs. Similarly, we determine the repeater needs for the longer nets (with unrepeated length greater than $4\,l_{rep}^{M3}$) using l_{rep}^{M6}. (Routes on M1 and M2 are usually too short to require repeaters).

While any repeater prediction scheme is, by its very nature, an approximation of the actual requirements, it does serve the

[1] Although we focus on improving the handling of repeaters required for the "common case" two-pin nets, the repeater needs of multi-pin nets can be estimated in a similar fashion. The net weighting method could also be improved with regard to multi-pin nets by using a length-based heuristic, as from [16].

purpose of allowing a designer to budget space for the appropriate numbers of repeaters along the eventual routings of the nets that will need them. The fine-tuning of these repeaters and their exact sizing can be carried out in ECO (Engineering Change Order) mode subsequent to the placement phase.

Although the repeater prediction scheme used by us captures the impact of layer assignment and valid inter-repeater distance ranges on repeater insertion, it considerably complicates the design of a net-weighting scheme to reduce repeater counts. We next describe how we overcome these complications.

3.1 Threshold-Based Net-Weighting

For each net, a net weight multiplier is created based on its net length and multiplied to the original net weight before the system of equations is solved in Kraftwerk. The net weight multiplier is a function of the current net length and the nearest threshold length at which a repeater would be deleted or inserted (i.e., $0.7\,l_{rep}^{M}$ or $1.4\,l_{rep}^{M}$ for the relevant layer *M* in our scheme). One can use any one of several different functional templates for this net-weighting function; however, its key features are that it has a value of one at a center point away from the repeater insertion and deletion thresholds, and that it gradually increases as one moves away from this center point, reaching a maximum value at the net length in which a repeater would be inserted or deleted (as shown in Figure 1). For all these templates, a single parameter, *viz*., the maximum possible value of net weight multiplier, is used to adjust the strength of the functions. Later in this paper, we present an empirical evaluation of several different functional templates.

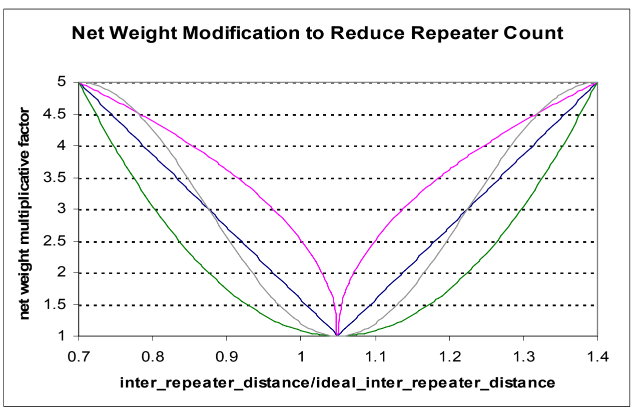

Figure 1. Possible functions for the net weight multiplier.

As can be seen in Figure 1, the functions consist of two halves. The right half discourages the net length from increasing beyond the threshold at which another repeater would be inserted. The left side encourages the net length to shrink beyond the next deletion threshold. This functional form is made symmetric to avoid introducing additional control parameters. It is replicated for each valid range of inter-repeater distances. When a net does not have any repeaters, the left side is ignored and set to one; since there are no repeaters to delete, encouraging the net to shrink further does not help reduce the repeater count.

Rapid changes in the net weight multipliers in successive iterations can cause the placement quality to degrade, and the net weight multiplier function itself may not be smooth. Therefore, we use exponential smoothing of the net weight multiplier based on the history of that multiplier over past iterations, in order to provide stability and promote convergence by ensuring that the multiplier does not change too rapidly. A smoothing constant of s $(0 \le s \le 1)$ implies that the new (smoothed) value of a net weight is given by $w'_{new} = (1-s)w'_{old} + s \cdot w_{new}$, where w'_{old} is the (smoothed) weight of that net in the previous iteration, and w_{new} is the unsmoothed weight for that net in the current iteration. Our scheme increases the average variation among different net weights at any time, and can potentially increase localized cell congestion after global placement. However, we faced no problems in the legalization of these regions even when our coarse legalizer FD-Mongrel was also using our net weighting scheme. Smoothing of the net weight multiplier also prevents instability issues in the initial iterations when net lengths are changing rapidly (because the multiplier asserts itself fully only if the length of a net remains near a repeater threshold over multiple iterations). Although we encountered no stability issue of this type, one could depreciate the net weight multipliers in the early iterations if needed, so that their full effect would be felt only in the later iterations when the spreading has stabilized.

In general, parameter tuning is not needed across different designs. The only control parameter, *viz*., the maximum multiplier value, is not very sensitive because our implementation of Kraftwerk automatically scales the spreading forces to ensure that the connectivity-induced attractive forces are balanced with the density-induced spreading forces. Consequently, changing our maximum net weight multiplier parameter merely changes the extent of wire length tradeoff between nets that are close to a threshold and the nets that are far from it, without impacting the spreading significantly. In other words, it changes only the spread (i.e. variance) of the connectivity forces, but not their mean value vis-à-vis the spreading forces.

3.2 Layer Transitions

The multiplier function must be handled carefully around the point at which the net switches layer pairs, so that the net weight multiplier remains continuous across different layer pairs, and net weighting produces its intended effect. Recall that the primary purpose of this multiplier is to encourage repeater deletions that are possible, and to discourage possible repeater insertions. If such an insertion or deletion is not possible because of a predicted layer transition, then the net weight should not be blindly increased even if a minimum or maximum inter-repeater distance threshold is being approached. This can happen in the transition region from the lower layer pair (say, M3-M4) to the higher layer pair (say, M5-M6). This is illustrated in Figures 2 and 3 using a linear net weighting function and generic wirelength values. The critical inter-repeater distances used in these two graphs are meant only to illustrate the two cases and do not represent the actual distances used in later experiments.

If a net is being modeled with the maximum possible repeaters for the lower pair, and it is approaching a net length at which it would switch over to the upper layer pair, then the weighting function should not increase even if it is approaching a M3-M4 maximum inter-repeater length threshold (see, for example, the dashed line labeled as “Not needed” in the right half of Figure 2). The net switches over to M5-M6 before it grows to a length at which another repeater would be inserted in a routing on M3-M4. Similarly, if a net has the fewest possible M5-M6 repeaters (in the sense that a further decrease in net length would cause the routing model to switch to using M3-M4), the net weight should not be increased for the purpose of deleting a repeater in the net weighting curve corresponding to M5-M6 (see the dashed line labeled as "Not Needed" in the left half of Figure 2).

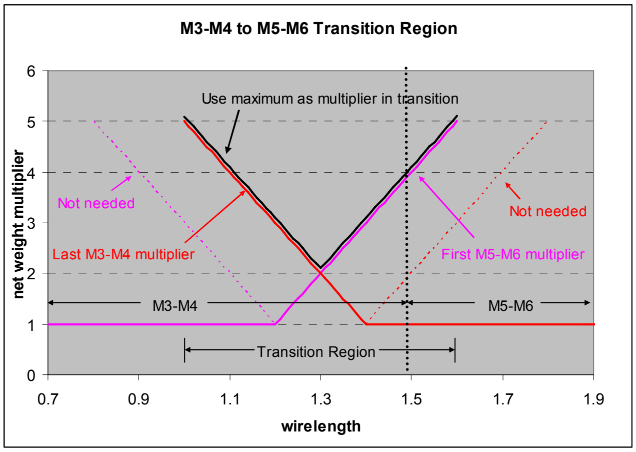

Figure 2. Overlapping M3-M4 deletion and M5-M6 insertion.

In Figure 2, the portions of the curves used for the last M3-M4 repeater deletion and the first M5-M6 insertion overlap directly; therefore, their max envelope is used. The other situation that can arise in the transition region (based on the amount of overlap between the last M3-M4 net weighting function and the first M5-M6 weighting function) is shown in Figure 3. In this figure, there is a certain amount of separation between the useful portions of the last M3-M4 curve and the first M5-M6 curve; so the multiplier is set to one in the intervening region. In both cases (*viz*., those illustrated in Figures 2 and 3), the insertion portion of the last M3-M4 curve and the deletion portion of the first M5-M6 curve are ignored and replaced by a value of one. The maximum of the resulting curves can simply be used in the transition region as a continuous net weight multiplier function in both cases.

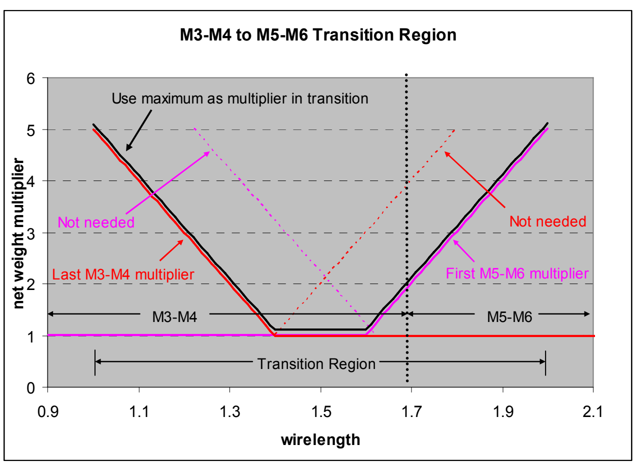

Figure 3. Overlapping M3-M4 insertion and M5-M6 deletion.

3.3 Implementation

In each iteration of Kraftwerk/MorePlace, we compute the net weight modifiers immediately after calling the repeater insertion/deletion procedure and before the system is solved for the new positions. This ensures that the net weights used by the solver are modified according to the most recent repeater configuration.

In FD-Mongrel, the calculation of the net weight modifiers is interleaved with the legalizer's iterations. The net weight modifiers are used as user-defined weights in Mongrel and are multiplied by the wirelengths in the gain function that determines which cells ripple-move in the fine grid from the most congested bin to the least congested bin. For convenience, these multipliers use the same net weighting function for both FD-Mongrel and Kraftwerk/MorePlace. However, exponential smoothing of the net weight multiplier is not needed in FD-Mongrel for stability[2]. Since the legalizer cannot work with virtual repeaters (unlike Kraftwerk/MorePlace), we instantiate the repeaters prior to passing the design to an FD-Mongrel iteration, but use the original nets to compute our net weight modifiers (in order to avoid fragmenting our netlist).

4. EXPERIMENTAL RESULTS

In our first set of experiments, we explored the extent of repeater count reduction possible using different net weighting functions. For each net weighting function, placements were created using Kraftwerk/MorePlace and FD-Mongrel with repeater reduction net weighting, and then legalized fully. A set of circuits from a recent microprocessor was used in these experiments, with inter-repeater distances corresponding to the 45 nm and 32 nm technology nodes as in [11]. The placements were generated using a 2.8 GHz Intel® Xeon™ server with 4 GB memory. In these experiments, exponential smoothing with a smoothing constant of 0.05 was applied to the net weight multipliers between iterations as discussed earlier.

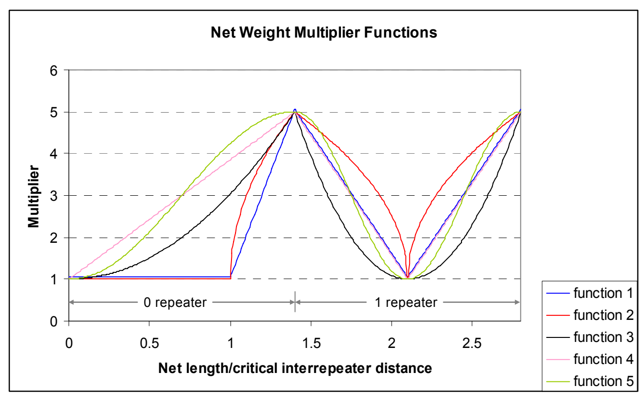

Figure 4. Possible functions for the net weight multiplier.

The five weighting functions shown in Figure 4 were compared in Table 2 for the testcases from Table 1 at the 32 nm node. The functions begin with an initial curve when no repeaters are present and continue to increase until the first maximum inter-repeater distance is reached. For one or more repeaters, the functions use somewhat of a v-shaped curve. The M3-M4 to M5-M6 transition region is handled as described earlier. Function 1 has a value of one up to the first critical inter-repeater distance; from there it

[2] Unlike global placement where the large flexibility available for each move necessitates an explicit history mechanism on net weights in order to create sufficient inertia to guard against extreme oscillations, the moves possible during legalization tend to be restricted enough that oscillations are avoided even without exponentially smoothed net weights.

increases linearly to the maximum multiplier at the threshold corresponding to the maximum inter-repeater distance. For one or more repeaters, the function has a linear v-shape with the midpoint having a net weight multiplier of one and the minimum and maximum thresholds having the maximum multiplier value. Function 2 is similar except that it increases and decreases using a square-root function scaled appropriately to the minimum and maximum values. Function 3 increases quadratically from a value of one at an inter-repeater distance of zero to its maximum at the first insertion threshold. For one or more repeaters, the v-shaped curve is quadratic. Functions 4 and 5 are similar to Function 3 except they use linear and sinusoidal curves respectively.

Table 1. Testcase details.

Testcase	Ckt_A	Ckt_B	Ckt_C	Ckt_D	Ckt_E
Cells	3978	4014	12312	13343	42127
Nets	4268	4384	13073	17685	42247

Table 2. Comparing function types at the 32nm node

Testcase	Function Type									
	Function 1		Function 2		Function 3		Function 4		Function 5	
	Δ rptrs	Δ L_{total}	Δ rptrs	Δ L_{total}	Δ rptrs	Δ L_{total}	Δ rptrs	Δ L_{total}	Δ rptrs	Δ L_{total}
Ckt_A	25%	4%	28%	4%	36%	3%	23%	3%	33%	2%
Ckt_B	44%	3%	38%	6%	43%	3%	35%	3%	40%	5%
Ckt_C	11%	5%	13%	6%	20%	1%	17%	1%	18%	3%
Ckt_D	13%	2%	12%	3%	15%	2%	16%	1%	17%	1%
Ckt_E	13%	-9%	15%	-9%	25%	-15%	16%	-9%	21%	-12%

The results of the different weighting functions are shown in Table 2. In this table, Δ rptrs is the percent decrease in the number of repeaters with net weighting as compared to the control flow in which neither of Kraftwerk/MorePlace or FD-Mongrel uses our net weight modifiers, and Δ L_{total} is the percent increase in the total wirelength with repeater reduction. Functions 1 and 2 seemed to perform poorly with the larger testcases, causing large wirelength increases. Function 4 was slightly better, but Functions 3 and 5 gave the most improvement in repeater reduction and the least wirelength increase. Overall, it appears that Function 3 gave the best results, particularly for the larger circuits which are more indicative of future trends. Therefore, we decided to use Function 3 for the rest of our experiments, calculating the net weight multipliers using a quadratic function of the inter-repeater distance.

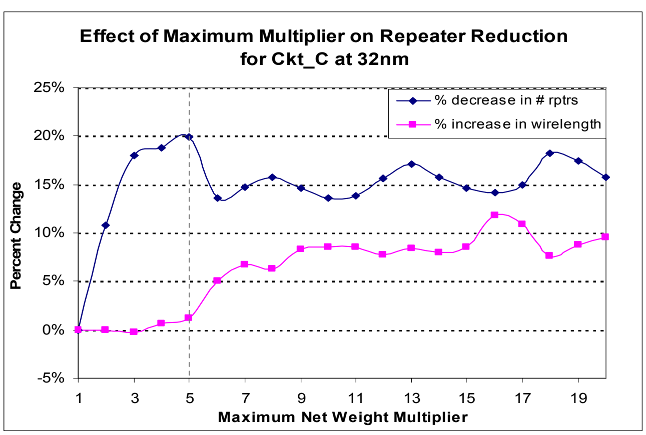

Figure 5. The effect of the maximum net weight multiplier.

We next studied the impact of our control parameter (*viz*. the maximum multiplier value) on the quality of the fully legalized placement for our median-sized testcase Ckt_C at 32 nm; the resulting change in repeater count reduction and wirelength is plotted in Figure 5. Smaller maximum values yield less repeater count reduction and smaller perturbations in the wirelength, while larger values cause more wirelength increase and better improvement in the repeater count reduction to a certain extent. However, if the value is too high, repeater count reduction decreases and more repeaters may actually be needed as wirelengths increase too much. A maximum value of 5 was chosen for rest of the experiments because the wirelength increase is low and the repeater count reduction is high at this point.

The results for placements scaled to the 45 nm and 32 nm nodes and generated using three different design flows are shown in Tables 3 and 4. All data is reported for fully legalized placements. Wirelengths are reported as half-perimeter measures. MP/FDM uses the original MorePlace and FD-Mongrel without repeater reduction. In MP-RR/FDM, MorePlace with repeater reduction is followed by regular FD-Mongrel. Finally, MP-RR/FDM-RR uses repeater reduction net weighting in both MorePlace and FD-Mongrel. All three design flows are followed by a fine-grained legalization. In these tables, # rptrs is the number of repeaters, L_{total} is the total wirelength, Δ rptrs and Δ L_{total} are respectively the percent decrease in repeater count and increase in total wirelength (compared to the MP/FDM results).

Table 3. Comparing repeater reduction at 32 nm.

Testcase	MP/FDM		MP-RR/FDM		MP-RR/FDM-RR		
	# rptrs	L_{total}	Δ rptrs	Δ L_{total}	Δ rptrs	Δ L_{total}	Time Overhead
Ckt_A	447	1.98E+07	10.7%	4.3%	36.2%	3.1%	25.7%
Ckt_B	431	1.75E+07	16.5%	1.9%	42.9%	3.4%	114.8%
Ckt_C	2869	8.62E+07	11.9%	1.3%	20.0%	1.2%	472.2%
Ckt_D	6304	1.57E+08	12.2%	0.3%	15.0%	1.8%	266.4%
Ckt_E	22984	5.42E+08	18.5%	-14.6%	25.3%	-14.5%	52.1%

Table 4. Comparing repeater reduction at 45 nm.

Testcase	MP/FDM		MP-RR/FDM		MP-RR/FDM-RR		
	# rptrs	L_{total}	Δ rptrs	Δ L_{total}	Δ rptrs	Δ L_{total}	Time Overhead
Ckt_A	124	1.99E+07	30.6%	1.1%	42.7%	2.2%	23.4%
Ckt_B	133	1.74E+07	30.1%	1.1%	63.2%	0.7%	114.0%
Ckt_C	1655	8.97E+07	26.0%	-1.9%	32.3%	-1.1%	20.6%
Ckt_D	3700	1.60E+08	22.7%	-1.3%	26.8%	-0.3%	188.1%
Ckt_E	11004	4.56E+08	12.8%	-2.6%	23.0%	-1.5%	-3.7%

Tables 3 and 4 show that, as expected, the best repeater reduction results are obtained when net weighting is applied to both MorePlace and FD-Mongrel. With this flow, 38% fewer repeaters are needed at 45 nm and 28% fewer repeaters at 32 nm. The wirelength deterioration is usually very low; in fact, the use of repeater reduction even decreases the total wirelength in several cases (because a design with fewer repeaters is more easily legalizable, thus causing less wirelength degradation after global placement). Even the results of the MP-RR/FDM design flow are better than the basic MP/FDM flow. In this flow, there is an average of 24% (14%) fewer repeaters at the 45 nm (32 nm) node respectively. The improved benefits of the MP-RR/FDM-RR flow over the MP-RR/FDM flow can be explained by the fact that the former continues active repeater reduction during the coarse legalization process. In contrast, a detailed analysis (omitted here due to space constraints) of the data for the MP-RR/FDM flow shows that some of the repeater reduction visible after the global placement phase is frittered away during coarse legalization in the absence of threshold-based net weights. Compared to coarse legalization, the smaller magnitude of the moves during fine-grained legalization does not impact the repeater gains significantly. Repeater count reduction does add a certain amount of overhead onto the runtime as shown in Tables 3 and 4 because of the additional computation and slower convergence. However, this overhead tends to be lower for the largest test cases because of the difficulty of fine-grained legalization with a larger number of repeaters in the baseline MP/FDM flow.

The impact of our threshold-based net weighting scheme on the wiring histogram of a design is illustrated using data for our median-sized testcase Ckt_C in Figure 6. It can be seen that repeater reduction decreases the length of nets near the repeater insertion threshold at the expense of nets with very short wirelengths (which do not require repeaters or have significant wire loads). This results in an increased aggregate of nets just before the wirelength where the first repeater would be added, just as one would expect. (Note that MorePlace causes a small perturbation in the wirelength histogram by itself near the first repeater insertion threshold, due to the additional quadratic forces in its attractive repeater force model).

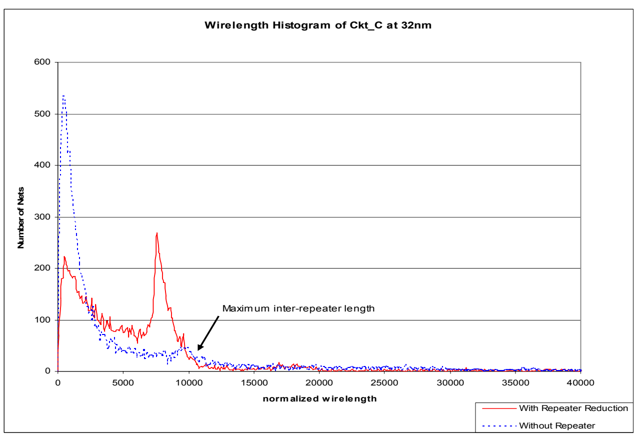

Figure 6. The wirelength histogram with repeater reduction.

Although this method was tested only with wire length driven placement, we are optimistic that our approach can be extended to timing-driven placement without deteriorating significantly under timing constraints. The rationale for our optimism is as follows. We found considerable robustness in the quality of our results even as different parameters were varied. Therefore, given that the majority of nets do not lie on timing-critical paths, one can guarantee that the most performance-sensitive timing-critical nets will not be unnecessarily lengthened (by forcing their "net weight multiplier" to its maximum value) and still allow for considerable flexibility in trading off net length between nets that are close to the repeater insertion thresholds and those non-critical nets that are not close to the thresholds. Furthermore, with lower repeater counts, our experiments demonstrate that legalization occurs with considerably less perturbation as measured by wirelength degradation and additional repeater requirements; this significant reduction in backend layout degradation that is a collateral benefit of our repeater reduction mechanism is especially important in ensuring timing closure after timing-driven placement.

5. CONCLUSIONS

Net weighting is useful not only in traditional timing- and power-driven placement, but also in reducing the number of repeaters needed in the design of future ICs. In this paper, we have shown that these repeater count reductions can be made without sacrificing placement quality. We have presented the mechanics of constructing a context-sensitive net weighting scheme that incorporates the effects of layer assignment and inter-repeater distance back-offs. Our scheme produced placements with significantly fewer repeaters, with only minor wirelength impact.

6. ACKNOWLEDGMENTS

The authors would like to thank T. Cao for providing a version of FD-Mongrel that accepts user defined net weights. They are also grateful to A. Chowdhary, K. Rajagopal, and B. Velikandanathan for providing support for the placement infrastructure, and D. Kirkpatrick and P. Cocchini for several useful discussions.

7. REFERENCES

[1] Bakoglu, H. B. Circuits, Interconnects and Packaging for VLSI. Addison-Wesley: Reading MA, 1990.

[2] Saxena, P., Menezes, N., Cocchini, P., and Kirkpatrick, D. A. Repeater Scaling and its Impact on CAD. *TCAD,* 2004, 23(4), 451-463.

[3] Cong, J., Kong T., and Pan, D. Z. Buffer Block Planning for Interconnect-Driven Floorplanning. *ICCAD* '99, 358-363.

[4] Tsay, R. S. and Koehl, J. An Analytic Net Weighting Approach for Performance Optimization in Circuit Placement. *DAC* '91, 620-625.

[5] Ren, H., Pan, D. Z., and Kung, D. S. Sensitivity Guided Net Weighting for Placement Driven Synthesis. *ISPD* '04, 10-17.

[6] Rajagopal, K., Shaked, T., Parasuram, Y., Cao, T., Chowdhary, A., and Halpin, B. Timing Driven Force Directed Placement with Physical Net Constraints. *ISPD* '03, 60-66.

[7] Obermeier, B. and Johannes, F. M. Temperature-Aware Global Placement. *ASP-DAC* '04, 143-148.

[8] Vaishnav, H. and Pedram, M. PCUBE: A Performance Driven Placement Algorithm for Low-Power Design. *Euro-DAC* '93, 72-77.

[9] Cheon, Y., Ho, P.-H., Kahng, A. B., Reda, S. and Wang, Q. Power-Aware Placement. *DAC* '05, to appear.

[10] Eisenmann, H., and Johannes, F. M. Generic Global Placement and Floorplanning. *DAC* '98, 269-274.

[11] Saxena, P., and Halpin, B. Modeling Repeaters Explicitly Within Analytical Placement. *DAC* '04, 699-704.

[12] Hur, S.-W., and Lillis, J. Mongrel: Hybrid Techniques for Standard Cell Placement. *ICCAD* '00, 165-170.

[13] Hur, S.-W., Cao, T., Rajagopal, K., Parasuram, Y., Chowdhary, A., Tiourin, V., and Halpin, B. Force Directed Mongrel with Physical Net Constraints. *DAC* '03, 214-219.

[14] Sigl, G., Doll, K., and Johannes, F. M. Analytical Placement: A Linear or a Quadratic Objective Function? *DAC* '91, 427-432.

[15] Akkiraju, N., and Mohan, M. Spec-based Flip-flop and Buffer Insertion. *ICCAD* '03, 270-275.

[16] Alpert, C. J., Hu, J., Sapatnekar, S., and Villarubia, P. Practical Methodology for Early Buffer and Wire Resource Allocation, *DAC* '01, 189-194.