SIEMENS
Ingenuity for life

Integrating Lagrangian Relaxation (LR) Gate Sizing in an Industrial Place-and-Route Flow

David Chinnery, Ankur Sharma
Siemens Digital Industries Software

Outline

- Motivation
- Overview of Lagrangian relaxation based gate sizing
- LR sizer runtime and speedup techniques
- Architecture and flow usage for the LR gate sizer with the Nitro-SoC industrial place-and-route (P&R) tool
- Results and conclusions

Motivation

Motivation for using LR based gate sizing

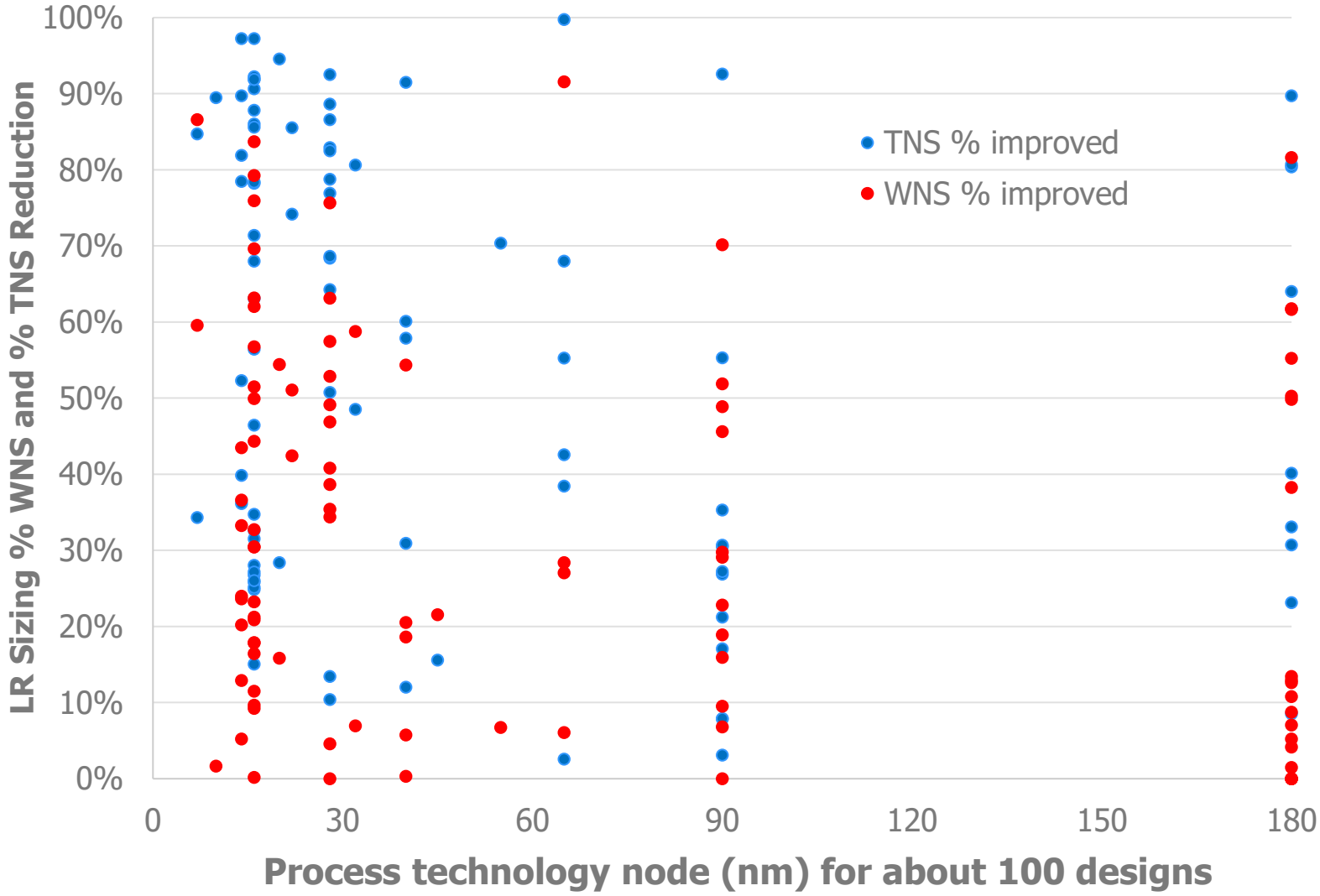
- Greedy sizing heuristics are sub-optimal, e.g.: +10% leakage vs. LR gate sizing
- State-of-the-art gate sizing results were shown on the ISPD 2013 gate-sizing contest with LR gate sizing by Flach et al. TCAD 2014.
- Sharma TCAD 2019 showed 15x speedup +2% leakage vs. Flach, sizing 884,427 gates in 31 minutes with 8 threads & a fitted Elmore RC wire model.
- TNS vs. leakage power trade-off was often poor at the start of the Nitro low power P&R flow, causing long WNS & TNS optimization runtimes.
  - **TNS** is **total negative slack** for setup timing violations summed across timing endpoints.
  - **WNS** is **worst negative slack** at any timing endpoint.

- **Primary goal:** Fast, high-quality gate-sizer for incremental sizing for WNS/TNS/area/power objectives to provide better timing vs. power trade-off.
- **Secondary:** modular, for use with other tools; distributed architecture support

Importance of gate sizing and Vth-swap vs. process technology (Pre-CTS fast opt experiment with leakage+area+timing LR objective)

- We see that gate-sizing can fix much of the outstanding setup timing WNS & TNS on the majority of the designs.
- Gate delays dominate at low supply voltage & high transistor threshold voltage (Vth) or long channel length.
- Wire delays dominate at high supply voltage in lower process nodes with higher wire RC values.
- LR sizing results also depend on how good the initial sizing is, and difficulty of fixing the violations (e.g. overconstrained paths, or with too great a logic depth).

Graph excludes the few data points where LR sizer degraded WNS or TNS.

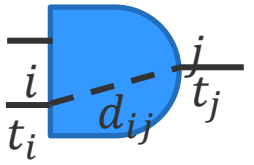


Overview of Lagrangian relaxation (LR) gate sizing

Lagrangian relaxation of the primal problem

- The primal problem is to minimize normalized power ($\hat{p}$) and area ($\hat{a}$):

$$\underset{\boldsymbol{x},\boldsymbol{t}}{\text{minimize}} \quad \sum_{g \in \text{cells}} \hat{p}_g(x_g) + \hat{a}_g(x_g)$$



$$\begin{aligned}
\text{subject to} \quad t_i + d_{ij}(\boldsymbol{x}) &\le t_j && \forall (i,j) \in \text{timing arcs} \\
t_k &\le T_k && \forall\, k \in \text{timing end points} \\
c_{\text{eff},i}(\boldsymbol{x}) &\le C_{\text{max},i}(\boldsymbol{x}) && \forall\, i \in \text{output timing nodes} \\
s_i(\boldsymbol{x}) &\le S_{\text{max},i}(\boldsymbol{x}) && \forall\, i \in \text{input timing nodes}
\end{aligned}$$

- **LR sub-problem:** primal objective + Lagrangian relaxed timing constraints:

$$\underset{\boldsymbol{x}}{\text{minimize}} \quad \sum_{g \in \text{cells}} \hat{p}_g(x_g) + \hat{a}_g(x_g) \quad + \sum_{(i,j) \in \text{arcs}} \lambda_{ij} \times \hat{d}_{ij}(\boldsymbol{x})$$

$$\begin{aligned}
\text{subject to} \quad c_{\text{eff},i}(\boldsymbol{x}) &\le C_{\text{max},i}(\boldsymbol{x}) && \forall\, i \in \text{output timing nodes} \\
s_i(\boldsymbol{x}) &\le S_{\text{max},i}(\boldsymbol{x}) && \forall\, i \in \text{input timing nodes}
\end{aligned}$$

  - Lagrange multipliers (λ_{ij}) weight the setup timing constraints.
  - Objectives are normalized vs. initial average cell area and power, and critical arc delay, to avoid having to rebalance them across different process technologies as units differ

Lagrangian relaxation problems to solve

- The Lagrangian relaxation sub-problem is a lower bound to primal problem:

$$\underset{\boldsymbol{x}}{\text{minimize}} \sum_{g \in \text{cells}} \hat{p}_g(x_g) + \hat{a}_g(x_g) \quad + \sum_{(i,j) \in \text{arcs}} \lambda_{ij} \times \hat{d}_{ij}(\boldsymbol{x}) \quad \leftarrow \text{ with optimal value } L_\lambda^* \text{ for this objective}$$

- Lagrangian dual problem to get the best lower bound:

$$\underset{\boldsymbol{\lambda}}{\text{maximize}} \quad L_\lambda^* - \sum_{k \in \text{ end points}} T_k \times \lambda_k$$

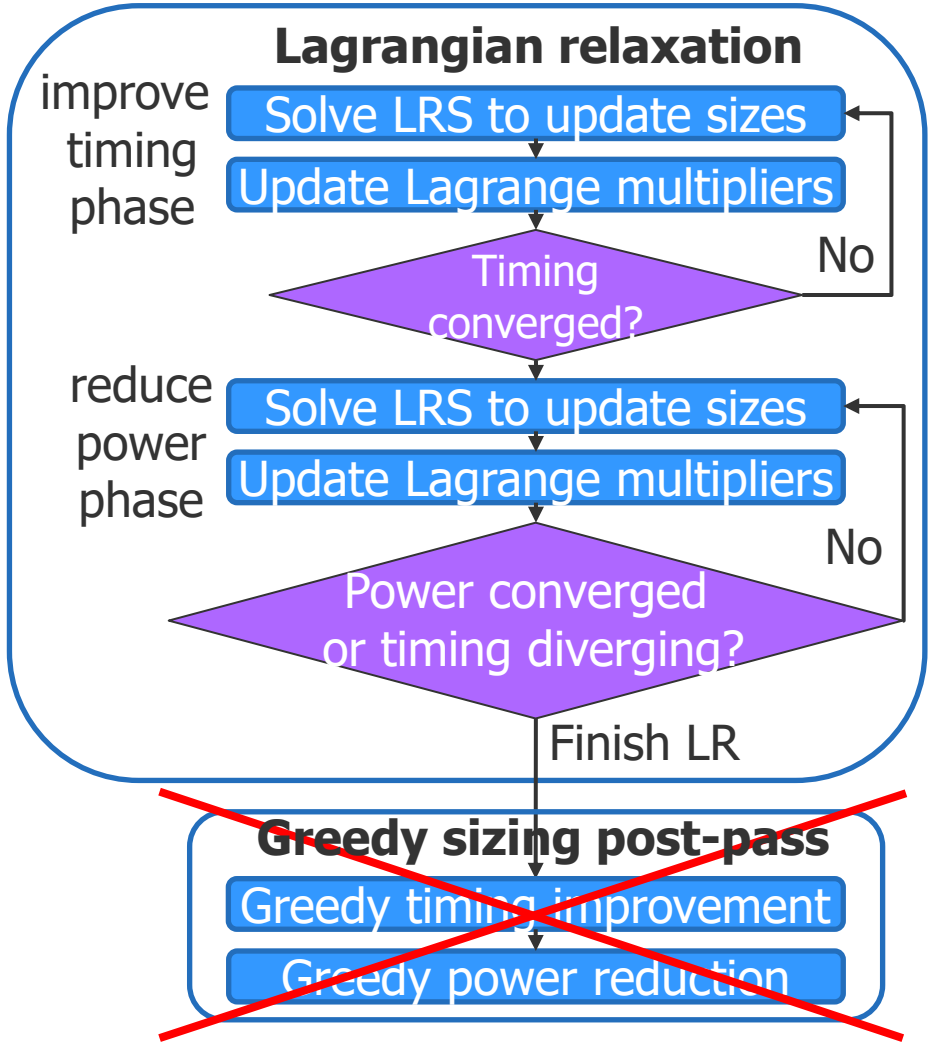
$$\text{subject to} \quad \sum_{\{u|(u,i) \in \text{arcs}\}} \lambda_{ui} = \sum_{\{v|(i,v) \in \text{arcs}\}} \lambda_{iv} \quad \forall\, i \in \text{nodes}$$

$$\sum_{\{u|(u,k) \in \text{arcs}\}} \lambda_{uk} = \lambda_k \quad \forall\, k \in \text{endpoints}$$

Karush-Kuhn-Tucker (KKT) optimality "flow" constraints

Lagrangian relaxation flow overview

- LR gate sizing has quite a few iterations of:
  - Resize gates to solve the Lagrangian relaxed sub-problem (LRS), for fixed multipliers.
  - Update the Lagrange multipliers based on the timing violations and to meet optimality conditions.
  - The improve timing and reduce power phases do similar resizing and multiplier updates. The phase affects exponent value in Lagrange multiplier update.
- Typically, there is also a greedy post-pass to fix minor timing violations and save power.
- We skip the greedy post-pass as it needs recalibration for timing accuracy + has extra runtime. Instead there's further sizing in the Nitro place-and-route flow.



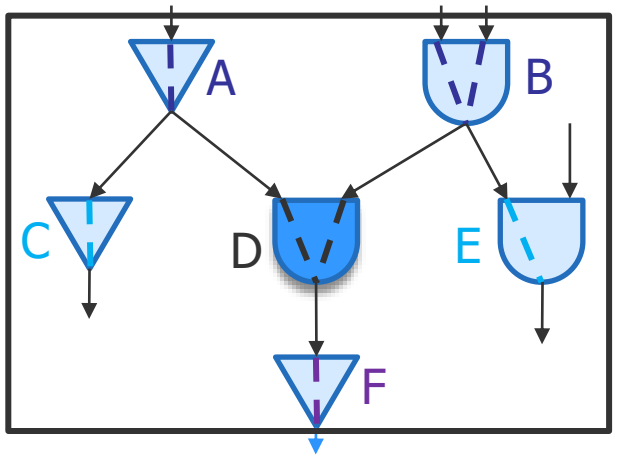
Heuristic to solve the LR sub-problem

$$\hat{p}_g + \hat{a}_g + \sum_{(i,j) \in \text{local arcs}(g)} \lambda_{ij} \times \hat{d}_{ij}$$

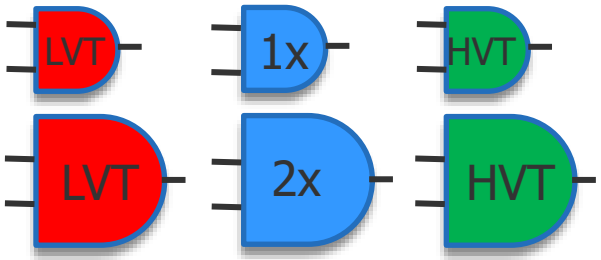
- Resize each gate to minimize the objective.
- While sizing a gate, check only its local neighborhood.
- λ_{ij} values are fixed as are neighboring gate sizes.
- Cell isn't resized if it locally worsens negative timing slack.
- Forward topological gate sizing, propagating arrivals.

- Choice of leakage or total power (leakage + dynamic) for the power objective, and optionally add area to objective.
  - Results later with these different objectives show the trade-offs.

Local neighborhood



Alternate cell choices



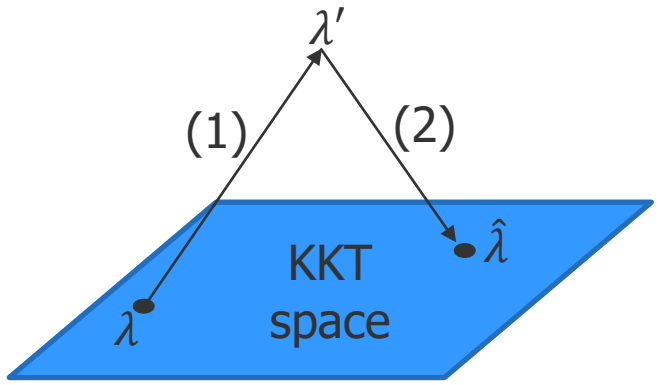
Heuristic to update Lagrangian multipliers (λ)

- Increase λ_{ij} if timing critical to penalize violations; else decrease the weight
- To update the Lagrange multipliers:
  (1) Adjust λ_{ij} disregarding the KKT constraints.

$$\lambda'_{ij} = \lambda_{ij} \times \left(1 - \frac{setup\_timing\_slack_{ij}}{average\_endpoint\_required\_time}\right)^{K}$$

  (2) Project the λ_{ij} to satisfy the KKT constraints.
  - Traverse the graph in reverse topological order and distribute the fanout arcs $\hat{\lambda}_{iv}$ to the fanin arcs $\hat{\lambda}_{ui}$:

$$\sum_{\{u|(u,i) \in \text{arcs}\}} \hat{\lambda}_{ui} = \sum_{\{v|(i,v) \in \text{arcs}\}} \hat{\lambda}_{iv}$$

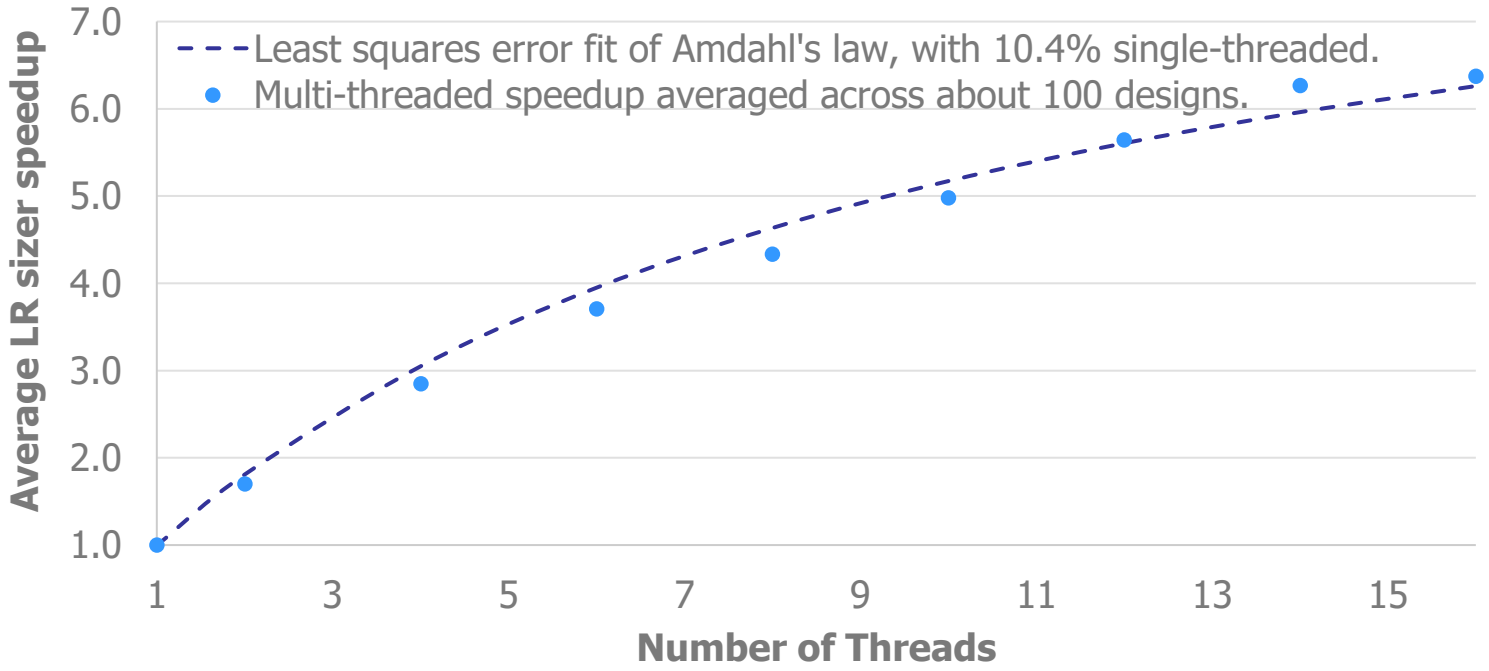


- During LR iterations, there are two phases for Lagrange multiplier updates:
  — First, in the timing improvement phase, Lagrange multipliers on critical (non-critical) arcs are updated with exponent $K = 4$ (1)
  — Then in power recovery phase they are updated with exponent $K = 1$ (4).

LR sizer runtime and speedup techniques

LR sizer runtime

- Good pre-CTS runtime achieved: average speed of 1.8 hours per million gates with 4 threads, and 0.8 hours with 16 threads.
- Post-CTS runtime is too slow, average speed of 5 hours with 4 threads.
- More enabled setup corners, timing arcs & timing graph nodes all increase runtime significantly more in post-CTS, and memory usage was also too high.



LR sizer speedup & memory reduction techniques

Speedup or Memory Reduction Technique	Speedup	Memory
Reduce memory by refactoring data structures, e.g., using struct for more compact data	23%	-21%
Switch from glibc malloc to tcmalloc 2.7, better memory recycling & thread allocation	22%	-19%
History-based adaptive library cell pruning	20%	0%
Skip non-critical (Lagrange multiplier λ < 1% of sum) fanin fanouts (sibling) arcs	19%	0%
Terminate LR power recovery phase when improvement less than 1% instead of 0.1%	13%	0%
Skip higher power library cells in power recovery	10%	0%
Cache nonlinear delay model (NLDM) cell library timing table coefficients	10%	0%
Multi-threaded data collection in Nitro	6%	0%
Reduce memory by not passing object names	2%	-3%

- Various techniques provided more than 3x speedup (the speedups multiply).
- 1% memory reduction gave almost 1% speedup.
- Algorithmic runtime improvements
  - Green: Reduction in iterations.
  - Blue: Reduction in number of library cell evaluations.
  - Yellow: Reduction in cell evaluation runtime.

History-based library cell (libcell) pruning reduced LR sizer runtime by 20%

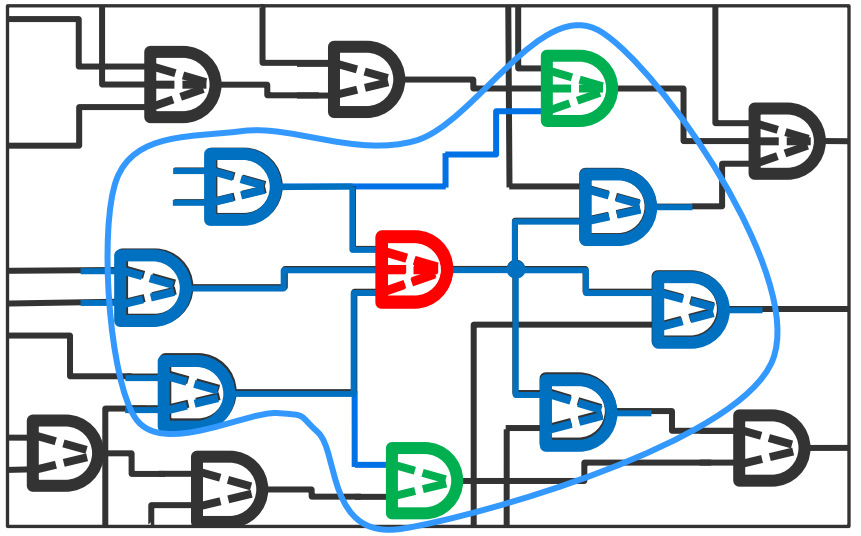
- As LR iterations progress, fewer cells are resized, and the order of best alternate library cells by cost stabilizes.
- Order libcells once every M iterations by the following cost function:

$$\left(\hat{p}_g + \hat{a}_g + \sum_{(i,j) \in \text{local arcs}(g)} \lambda_{ij} \times \hat{d}_{ij}\right) + 10^6 \times local\_slack\_change$$

  — First ordering happens when less than 10% of cells are resized.
- Evaluate lowest cost 20% (at least 2) libcells until next ordering.
- M is adapted on a per-cell basis depending on the jump in the optimal libcell, in an ordered array.
  — If jump ≤ 20%, ordering can be less frequent. Increment $M = M + 1$.
  — Else, decrement $M = \max\{M - 1, 2\}$.

Skipping non-critical sibling arcs reduced LR size runtime by 19%

- The number of fanin fanouts (siblings in green) can be many, and it increases the runtime for local timing analysis.
- Resizing the current gate has only a second order impact on sibling arc delay.
  - 1st order impact on fanin, self, and fanout arcs.



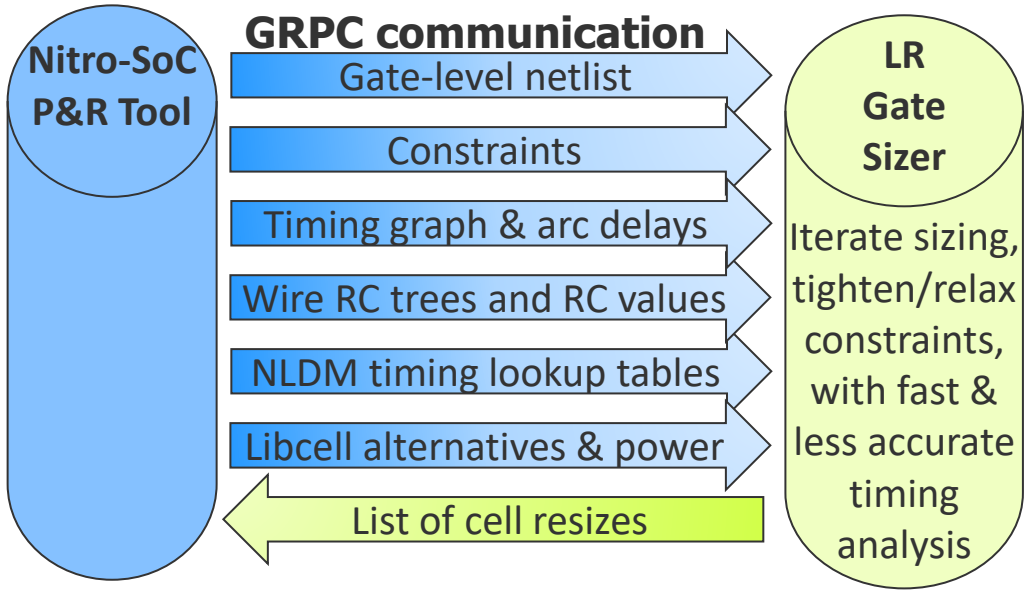
- Non-timing-critical sibling arcs have small Lagrange multipliers and contribute little change in the cost, so they can be skipped.

- Sum up Lagrange multipliers for every local-arc.
- Skip sibling arcs that contribute < 1% of the multiplier sum.

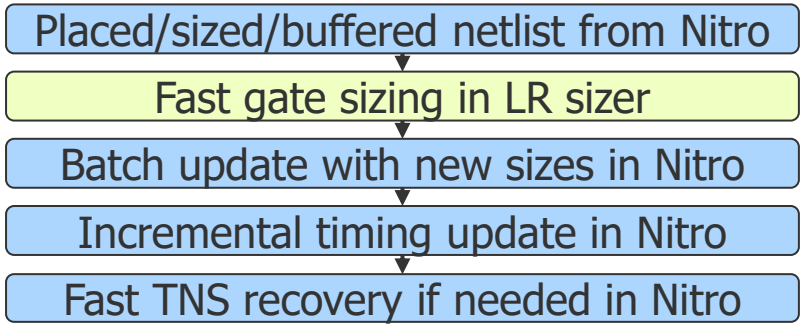
Architecture and flow usage for the LR gate sizer with the Nitro-SoC industrial P&R tool

Architecture with a modular standalone LR gate sizer

- We use Google's remote procedure call (GRPC) and protocol buffer framework to serialize and transfer data between Nitro and LR sizer + to send commands.
- The timing graph, arrival/required times, and so forth, are sent from Nitro to the LR sizer.
- Nitro quickly precalculates power for alternate libcells in each cell's context and this is also send to LR sizer.
  - Ignore cell's short circuit power dependence on varying input slew as it is only about 5% of cell power.
- Fast and more optimal gate-sizing is done in LR sizer.
- Then updated cell sizes are sent back to Nitro, which then continues the rest of the place-and-route flow.



Basic flow usage of the LR gate sizer

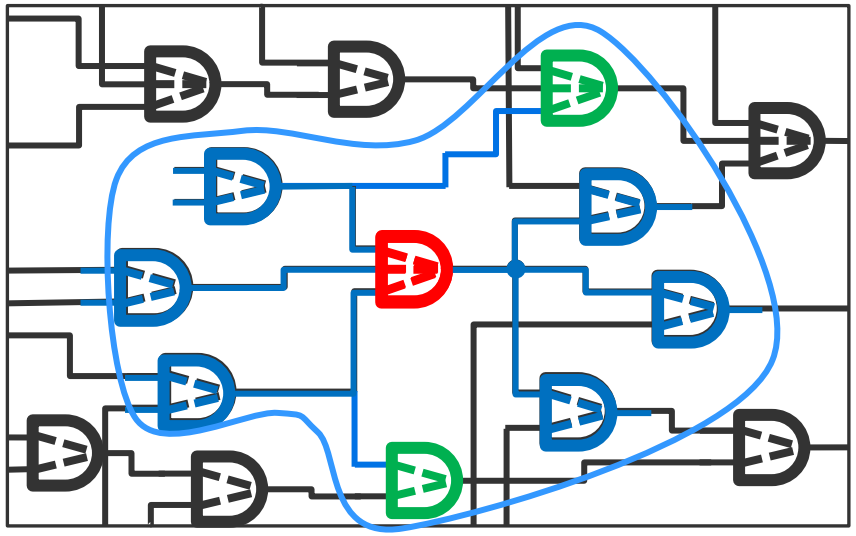


LR timer speedups and timing accuracy

- We needed a very fast timer as LR needs multiple iterations to converge.
  - We averaged 23 LR iterations: 13 for timing improvement and 10 for power reduction.
- NLDM lookup tables + Elmore wire RC delay model assuming no rerouting with fixed cell positions, no connectivity changes, and fixed wire RC trees.
- Inaccurate vs. composite current source (CCS) timing analysis with Arnoldi delay propagation in Nitro, so we fit coefficients to improve accuracy.
  - LR sizer's initial WNS correlates within 10ps of Nitro.
  - There can be timing degradation in Nitro after LR sizing, due to LR timer inaccuracies.
  - Advanced on-chip variation (AOCV) and clock reconvergence pessimism removal (CRPR) are not modeled in the LR timer, worsening post-CTS timing correlation.
- LR timer's analysis is more than an order of magnitude faster vs. Nitro.
- No updates in Nitro until all LR sizing finished, then just one Nitro timing update with the final sizes sent from the LR sizer.

Lightweight local timing graphs for LR sub-problem

- The main timing graph is read-only during local analysis for a resize.
- Small portion of the main timing graph
  - Target cell (red) + fanouts and fanins (blue) + fanins' fanouts (siblings in green)
  - Only node/arc "states" are duplicated
  - Fixed arrival times and input slews at the local inputs
  - Fixed required times and load capacitance at the local outputs



- After picking the best gate size, copy timing from local graph back to main.
- Multi-threaded with multiple local timers to resize multiple gates in parallel.
- Mutual exclusion edges (MEEs) enforce gate size ordering to avoid read/write collisions on the main timing graph when we update from the local timers.

Some of the major steps in the Nitro P&R flow: place stage, where we may call LR once or twice

Excellent LR sizing TNS improvement + power reduction was achieved early in pre-CTS, so enable it there.

Place stage:

- Global placement; fast sizing/buffering; high fanout synthesis; placement legalization; global route nets.
- ✦ WNS/TNS/area/power optimization with **LR gate sizer**
- WNS/TNS/DRV optimization – fast; then accurate but slower
- Leakage + dynamic power minimization, cell area may not increase.
- WNS/utilization/routing-congestion okay? If yes, then skip to last past the steps in red.
- Incremental global placement, placement legalization, and global route nets
- ✦ WNS/TNS/area/power optimization with **LR gate sizer** + WNS/TNS/DRV optimization – fast
- WNS/TNS/DRV optimization – accurate
- If congestion/utilization high, do: leakage + dynamic power minimization, cell area may not increase.
- Placement legalization and global routing update.
- WNS optimization + legalization & global routing update. Rollback subset of changes if WNS is degraded.

Incremental global placement significantly perturbs the timing, reducing the effectiveness of LR gate sizing, if the optional steps in red are performed. We can add a second LR gate sizer invocation to respond to this.

Some of the major steps in the Nitro P&R flow: clock stage where may call LR once, & route stage

Clock stage:

- Clock tree synthesis (CTS); detail route clock trees; placement legalization; and update routing.
- ✦ WNS/TNS/area/power optimization with **LR gate sizer**
- WNS/TNS/DRV optimization – fast; then accurate but slower
- Leakage + dynamic power minimization, cell area may not increase.
- Placement legalization and global routing update.
- WNS/TNS/DRV/hold optimization – accurate

Clock tree synthesis also significantly perturbs the timing, reducing the effectiveness of LR gate sizing. We can add a post-CTS LR gate sizer invocation to respond to this.

Route stage:

- Detail route signal nets
- WNS/TNS/DRV/hold optimization – accurate
- Leakage + dynamic power minimization, cell area may not increase.
- Placement legalization and detailed routing update.
- WNS/TNS/DRV/hold optimization – accurate + legal placement + update detailed routes.

Results and conclusions

How to analyze noisy flow results?

- Flow-step level and end-of-flow-stage noise in results is high.
- Averaging violations' relative differences doesn't work, as 1us → 2us TNS degradation is much worse than 1ps → 2ps, though both are 100% worse.
- Instead, violations are summed over designs to compare, reducing the weight of degradations vs. small values. It can still be dominated by a large outlier.
  - In practice, large outliers will be debugged and often fixes are made to address them.
- So we also compare the ratio of improved to worse results for a given metric.
- Full flow results are reported vs. the high-effort low-power Nitro P&R flow, which provides a strong baseline to compare against.
- Results here are for 67 designs with process technology from 180 to 7nm.
- Results that are worse than baseline will be shown in red.

Pre-CTS and post-CTS sizing results on 67 designs

Metric type	Sum Difference Over Designs				Mean Relative Diff.		
Pre-CTS Fast Opt. Step LR Sizer Objective	WNS	TNS	Max Slew Viol. Sum	Illegal Cells	Cell Area	Leakage Power	Total Power
timing, leakage, area	-0.7%	-18.0%	-20.5%	16.6%	-2.49%	-11.01%	-3.06%
timing, leakage	-0.7%	-18.5%	-25.9%	39.2%	-0.23%	-11.42%	-1.69%
timing, total power, area	-0.5%	-19.1%	-19.2%	11.5%	-3.00%	3.22%	-3.64%
timing, total power	-0.5%	-19.8%	-22.1%	21.5%	-2.20%	1.93%	-3.55%

Ratio of number of better results to worse results for a metric, out of 67 designs.

Pre-CTS Fast Opt. Step LR Sizer Objective	WNS	TNS	Max Slew Viol. Sum	Illegal Cells	Cell Area	Leakage Power	Total Power
timing, leakage, area	42:19	43:24	36:23	21:46	52:15	52:10	56:11
timing, leakage	44:16	44:22	36:22	8:59	39:28	51:10	53:14
timing, total power, area	42:17	47:19	31:26	22:45	52:15	39:22	57:10
timing, total power	42:20	47:20	30:28	16:51	47:20	36:25	56:11

- There's 11% to 14.5% leakage power reduction in both pre/post-CTS with the LR leakage objective.
- LR sizer + fast opt vs. fast opt reduces TNS by 18.0 to 19.8% in pre-CTS and 6.2 to 9.5% in post-CTS.
- Total power reduction ranges from 0.9% to 3.6%, more with the LR total power objective, less in post-CTS.
- Slew violations are reduced vs. baseline with a fast optimization post-pass to fix violations of TNS, input slew, and max load capacitance.
- The area objective is needed to reduce illegal cell placement and subsequent displacement for legalization. Including the area objective only reduces leakage power savings by about 1%.

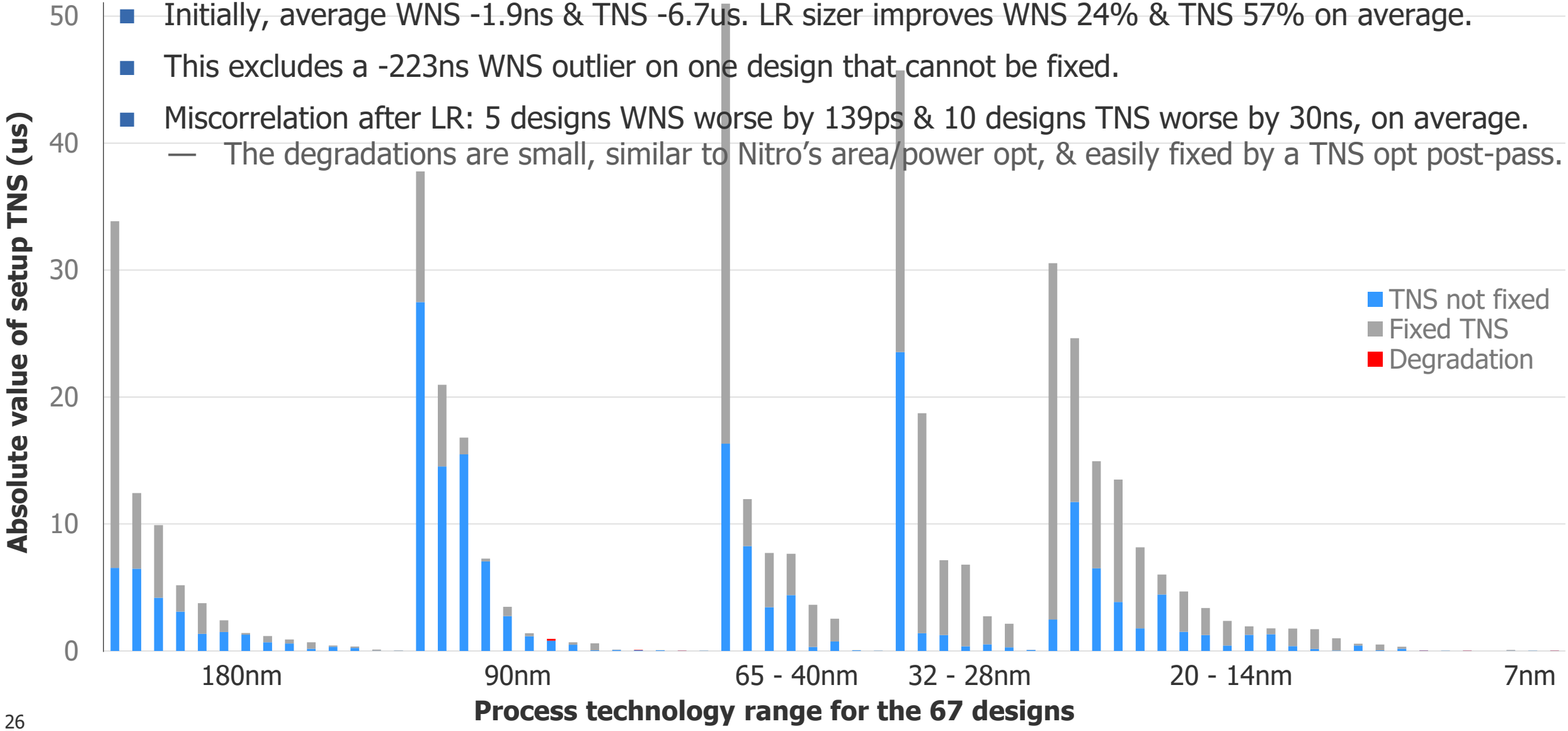
Metric type	Sum Difference Over Designs				Mean Relative Diff.		
Post-CTS Fast Opt. Step LR Sizer Objective	WNS	TNS	Max Slew Viol. Sum	Illegal Cells	Cell Area	Leakage Power	Total Power
timing, leakage, area	0.4%	-6.2%	-47.7%	25.4%	-1.38%	-14.51%	-0.88%
timing, total power, area	0.1%	-9.5%	-32.3%	11.5%	-2.22%	-0.38%	-1.75%

Ratio of number of better results to worse results for a metric, out of 67 designs.

Post-CTS Fast Opt. Step LR Sizer Objective	WNS	TNS	Max Slew Viol. Sum	Illegal Cells	Cell Area	Leakage Power	Total Power
timing, leakage, area	24:31	28:38	58:3	15:52	55:12	55:7	51:16
timing, total power, area	28:30	35:31	52:9	22:45	60:7	42:17	60:7

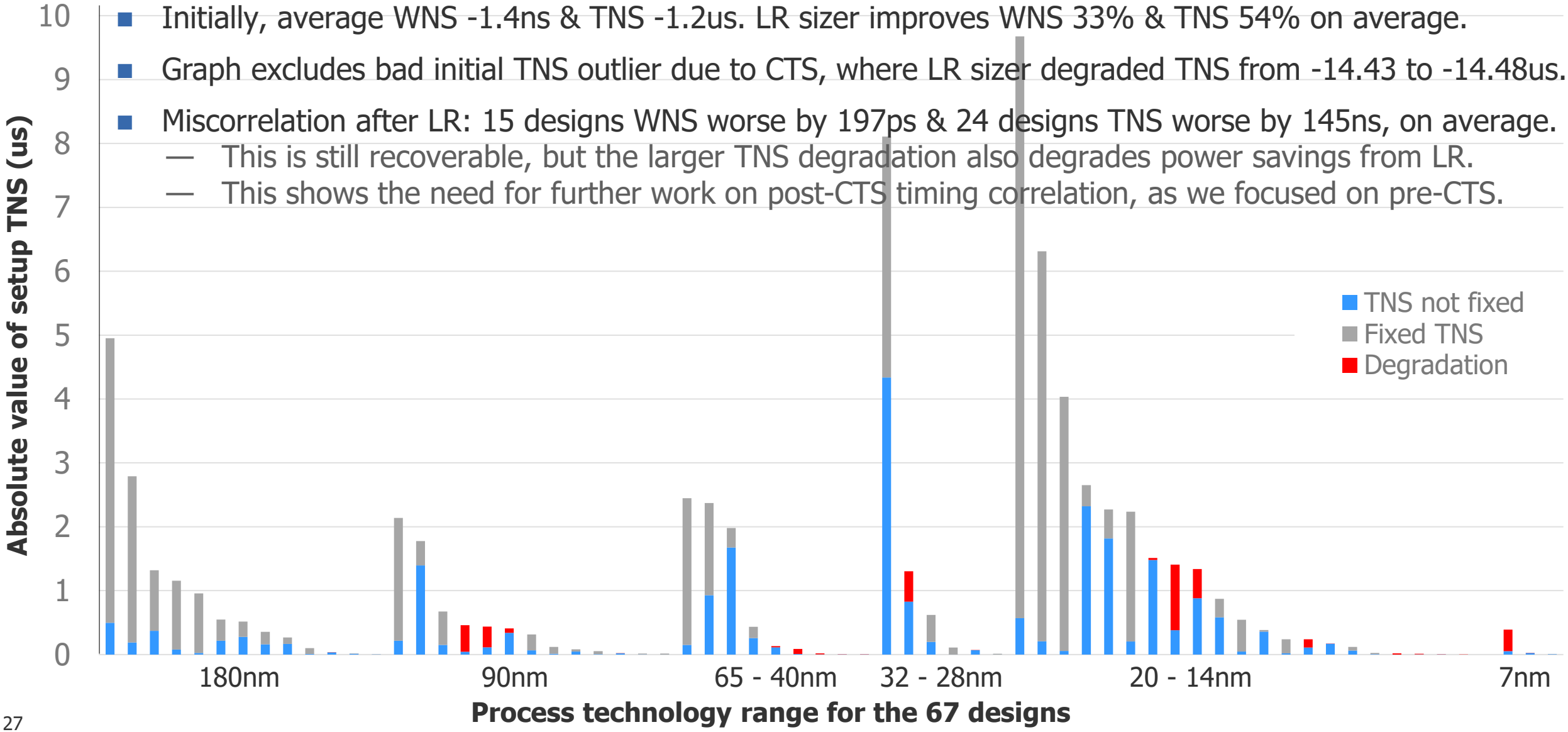
LR sizer improvement from initial TNS at pre-CTS fast opt step for the LR leakage + area objective

- Initially, average WNS -1.9ns & TNS -6.7us. LR sizer improves WNS 24% & TNS 57% on average.
- This excludes a -223ns WNS outlier on one design that cannot be fixed.
- Miscorrelation after LR: 5 designs WNS worse by 139ps & 10 designs TNS worse by 30ns, on average.
  - The degradations are small, similar to Nitro's area/power opt, & easily fixed by a TNS opt post-pass.

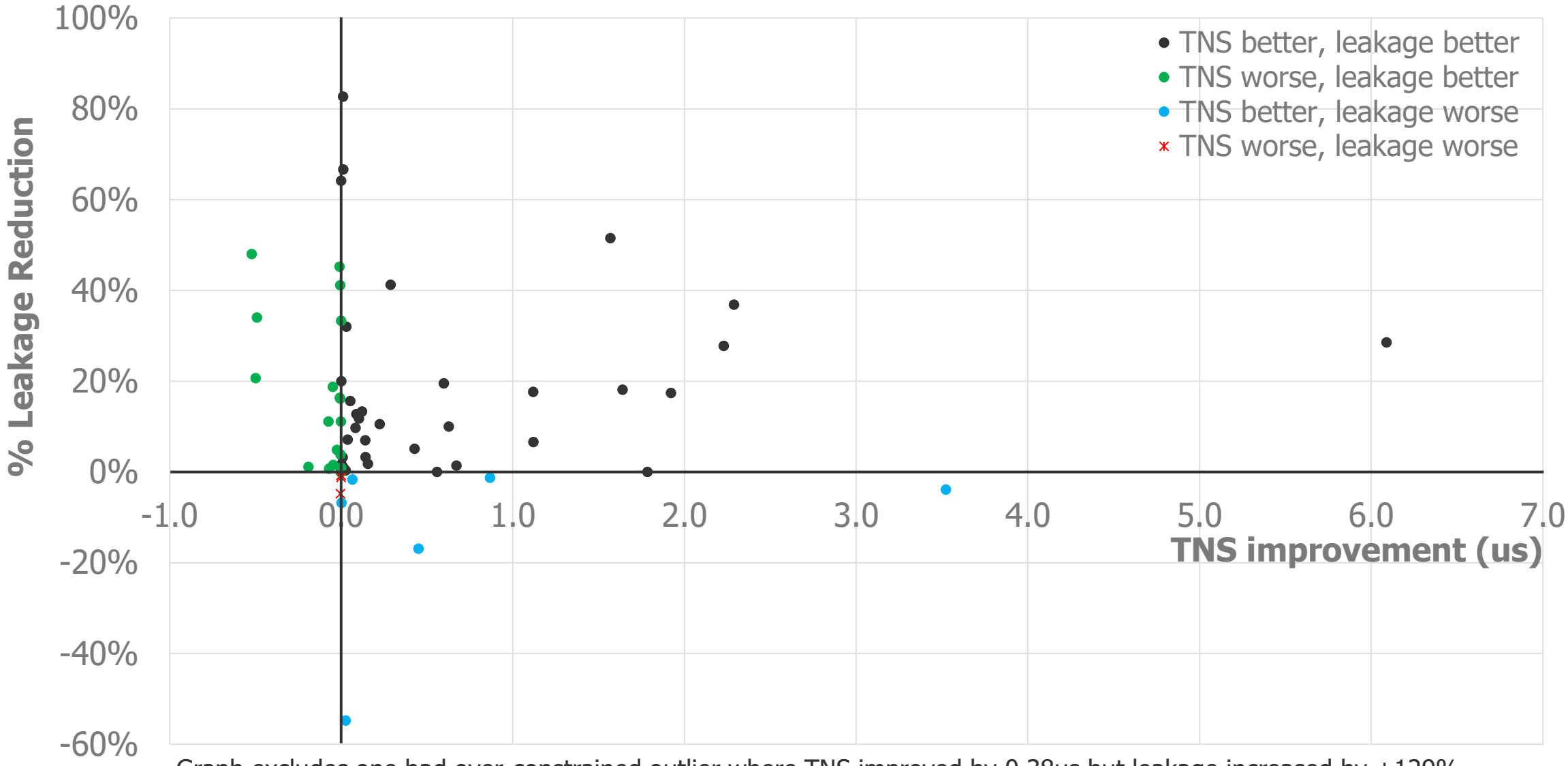


LR sizer improvement from initial TNS at post-CTS fast opt step for the LR leakage + area objective

- Initially, average WNS -1.4ns & TNS -1.2us. LR sizer improves WNS 33% & TNS 54% on average.
- Graph excludes bad initial TNS outlier due to CTS, where LR sizer degraded TNS from -14.43 to -14.48us.
- Miscorrelation after LR: 15 designs WNS worse by 197ps & 24 designs TNS worse by 145ns, on average.
  - This is still recoverable, but the larger TNS degradation also degrades power savings from LR.
  - This shows the need for further work on post-CTS timing correlation, as we focused on pre-CTS.

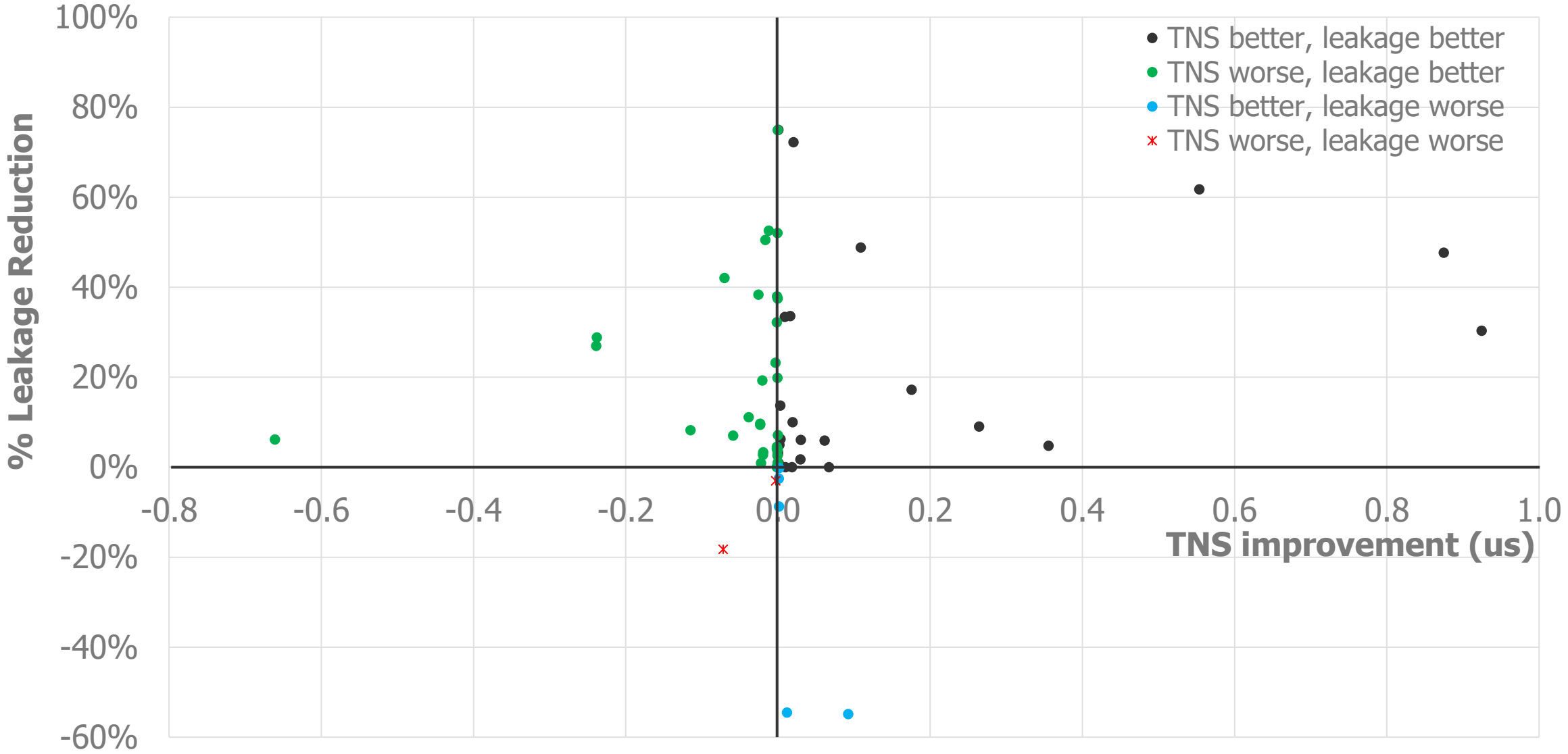


Pre-CTS LR sizer with leakage + area objective then Nitro fast optimization vs. Nitro fast optimization



Graph excludes one bad over-constrained outlier where TNS improved by 0.38us but leakage increased by +129%.

Post-CTS LR sizer with leakage + area objective then Nitro fast optimization vs. Nitro fast optimization



Full flow experiments enabling LR sizer at different points in pre-CTS place and post-CTS clock stages

	Flow	Sum Difference		Mean Relative Difference		
Sizer Flow Experiment	Stage	WNS	TNS	Area	Leakage	Tot. Pow.
1x leakage pre-CTS	place	-0.4%	-1.4%	0.07%	-3.28%	-0.10%
No LR sizer in post-CTS	clock	-3.4%	52.5%	-0.10%	-2.27%	-0.30%
	route	-17.9%	44.6%	-0.10%	-2.38%	-0.45%
2x leakage pre-CTS	place	3.1%	2.2%	0.18%	**-5.13%**	0.01%
No LR sizer in post-CTS	clock	-0.3%	27.9%	-0.07%	-2.92%	-0.21%
	route	-16.9%	23.6%	-0.12%	-2.88%	-0.41%
1x leakage pre-CTS	place	-0.2%	-2.1%	0.09%	-3.16%	-0.08%
1x leakage post-CTS	clock	-1.3%	48.1%	-0.50%	-6.01%	-0.93%
	route	11.0%	50.1%	-0.38%	**-4.89%**	-0.83%
No LR sizer in pre-CTS	place	0.9%	2.7%	-0.02%	0.05%	0.02%
1x leakage post-CTS	clock	-1.0%	6.1%	-0.60%	**-6.59%**	-1.01%
	route	-17.3%	-2.3%	-0.36%	-4.59%	-0.64%
1x total_power pre-CTS	place	4.7%	0.6%	-0.10%	-0.27%	-0.01%
No LR sizer in post-CTS	clock	-0.9%	36.1%	-0.23%	-1.20%	-0.31%
	route	-17.5%	25.5%	-0.14%	-1.50%	-0.48%
2x total_power pre-CTS	place	7.0%	4.9%	**-0.22%**	-0.58%	-0.25%
No LR sizer in post-CTS	clock	-2.6%	8.9%	-0.23%	-1.54%	-0.34%
	route	-19.8%	-0.1%	-0.23%	-1.71%	-0.45%
1x total_power pre-CTS	place	3.9%	-3.8%	-0.10%	-0.31%	0.04%
1x total_power post-CTS	clock	-4.5%	1.8%	-0.84%	-2.29%	**-1.03%**
	route	-18.3%	2.7%	**-0.62%**	-2.54%	**-0.94%**
No LR sizer in pre-CTS	place	0.6%	3.6%	-0.01%	-0.17%	-0.07%
1x total_power post-CTS	clock	-4.2%	35.1%	**-0.88%**	-3.32%	-1.03%
	route	-19.2%	23.0%	-0.57%	-2.32%	-0.59%

- Best results shown in **blue**.
- Best power reduction is with LR sizer in pre-CTS & post-CTS.
- Average 4.9% less leakage and 0.94% less total power for the corresponding LR objectives.
- Clock & route TNS% dominated by outlier, baseline -8.5us worsens to -14.2 to -16.8us in some runs.
- Setup timing impact is noisy. Roughly neutral for runs where we only use LR in post-CTS.
- Impact on other metrics such as hold timing and design rule violations is roughly neutral.
- LR objective includes area in all runs, to reduce displacement.

Ratio of number of better results to worse for a metric for 67 designs.

Sizer Flow Experiment	Stage	WNS	TNS	Area	Leakage	Tot. Pow.
1x leakage pre-CTS	place	**35:21**	37:23	38:29	41:18	39:28
No LR sizer in post-CTS	clock	34:23	27:32	36:30	40:17	40:27
	route	26:22	23:25	39:28	41:18	40:27
2x leakage pre-CTS	place	31:27	**39:21**	39:28	**49:12**	37:30
No LR sizer in post-CTS	clock	30:26	27:32	42:25	47:13	37:30
	route	21:26	23:24	39:28	42:17	36:31
1x leakage pre-CTS	place	35:22	35:25	37:30	40:19	37:30
1x leakage post-CTS	clock	29:27	24:34	47:20	52:8	44:23
	route	21:28	19:31	44:23	51:9	45:22
No LR sizer in pre-CTS	place	25:25	29:24	28:28	20:26	26:31
1x leakage post-CTS	clock	23:33	25:32	57:10	**56:4**	**50:17**
	route	23:24	22:25	52:15	**55:6**	44:23
1x total_power pre-CTS	place	30:29	36:24	39:28	29:29	36:31
No LR sizer in post-CTS	clock	31:27	24:34	37:30	30:27	35:32
	route	23:25	21:28	38:28	32:26	36:31
2x total_power pre-CTS	place	32:28	35:28	**48:19**	39:22	40:26
No LR sizer in post-CTS	clock	29:31	28:32	39:28	38:21	37:30
	route	**27:21**	22:26	39:27	36:24	40:27
1x total_power pre-CTS	place	31:28	37:23	42:25	32:27	36:30
1x total_power post-CTS	clock	29:29	24:35	51:16	45:15	48:19
	route	25:25	19:32	47:20	44:16	43:24
No LR sizer in pre-CTS	place	27:22	26:28	28:27	26:22	27:29
1x total_power post-CTS	clock	**35:22**	**29:29**	**62:5**	50:8	**50:17**
	route	**27:21**	23:25	**53:14**	46:15	**45:22**

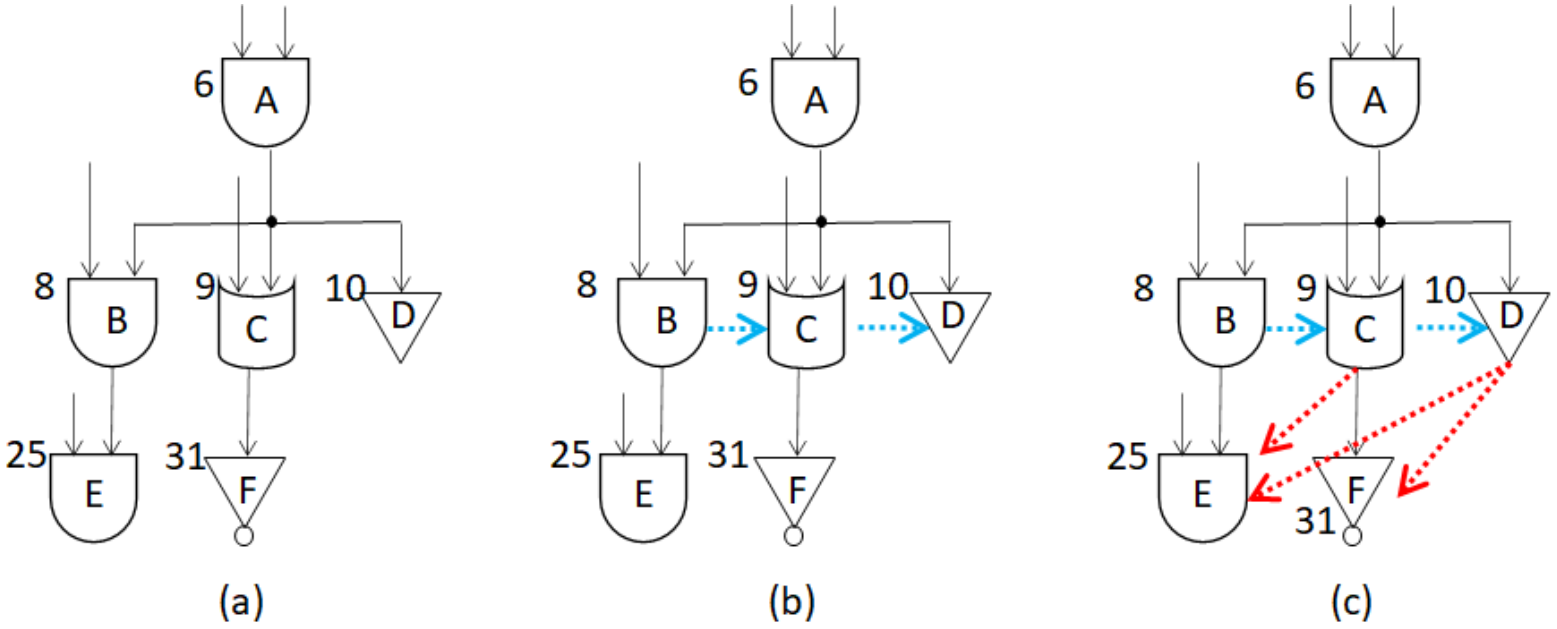
Conclusions

- LR gate sizing fixed more than half the setup timing violations in both pre-CTS and post-CTS, augmenting other optimization techniques.
  - At the fast optimization flow step, 10 to 20% less TNS with 11 to 14% less leakage or 1% to 3% lower total power, & 1% to 3% less cell area.
  - Achieved 3x LR sizer speedup to resize 10^6 gates in 0.8 hours pre-CTS with 16 threads.
  - NLDM timing analysis with Elmore RC delays and fitting coefficients provided very fast timing analysis so that LR gate sizing can do the multiple iterations it needs to converge.
- Full flow reductions of 5% leakage, 1% total power, 0.6% cell area can be achieved, with roughly neutral impact on other metrics.
  - Timing improvements not seen on average in the full flow, but LR gate sizing reduces subsequent optimization needed in more accurate but slower flow steps.
  - Runtime impact on flow was about neutral for a single invocation of LR sizer in pre-CTS.
- Good full flow results require flow tuning and debugging outliers.
- Further work was needed for post-CTS usage, to reduce the LR sizer runtime and improve timing correlation.

Backup slides

Additional mutual exclusion edges (MEEs) to siblings for deterministic multi-threaded results

- MEEs (in blue) provided multi-thread safe resizing for the LRS solver, avoiding simultaneous resizing of sibling cells to prevent write-write collisions on the main timing graph.



- However, read/write collisions still could occur due to sizing of a cell and its sibling's fanouts. For example, after resizing cell C, we update its fanin A and sibling B and D arrival times and slews in main timing graph, which impacts cell E. So what timing values E sees depends on order of completing sizing cells C and E.
- We introduce additional MEEs (in red) to enforce an order among sibling cells and their fanouts. For example, C and D are resized before E and F.

Timer calibration enhancements

- LR timer is fast but approximate – based on NLDM look-up tables.
- Correction factors are calibrated for better timing correlation.
- Five correction factors were proposed previously to calibrate effective cap, cell and net delays, and cell and net slews.
- To increase timing accuracy for additional issues seen on industrial designs, we added:
  - A pin capacitance correction factor to calibrate pin cap and total capacitance, to account for the Miller effect on input pin capacitance.
  - A slew scaling factor to sharpen the slew on nets going from a higher to lower voltage domain.

Full flow results with hold timing violations worst (WHS) and total (THS) and results for LR leakage + area objective run only in post-CTS

	Flow	Difference in Sum				Mean Relative Difference		
Sizer Flow Experiment	Stage	WNS	TNS	WHS	THS	Area	Leakage	Tot. Pow.
leakage pre-CTS	place	-0.4%	-1.4%			0.07%	-3.28%	-0.10%
No LR sizer in post-CTS	clock	-3.4%	52.5%	-12.6%	-13.6%	-0.10%	-2.27%	-0.30%
	route	-17.9%	44.6%	-62.9%	-35.5%	-0.10%	-2.38%	-0.45%
2x leakage pre-CTS	place	3.1%	2.2%			0.18%	-5.13%	0.01%
No LR sizer in post-CTS	clock	-0.3%	27.9%	-9.8%	-23.3%	-0.07%	-2.92%	-0.21%
	route	-16.9%	23.6%	-68.0%	-57.6%	-0.12%	-2.88%	-0.41%
leakage pre-CTS	place	-0.2%	-2.1%			0.09%	-3.16%	-0.08%
leakage post-CTS	clock	-1.3%	48.1%	-21.0%	-32.3%	-0.50%	-6.01%	-0.93%
	route	11.0%	50.1%	-56.7%	-55.1%	-0.38%	-4.89%	-0.83%
No LR sizer in pre-CTS	place	0.9%	2.7%			-0.02%	0.05%	0.02%
leakage post-CTS	clock	-1.0%	6.1%	-22.7%	-95.1%	-0.60%	-6.59%	-1.01%
	route	-17.3%	-2.3%	-81.6%	-88.7%	-0.36%	-4.59%	-0.64%
1x leakage pre-CTS	place	3.2%	-0.6%			-0.20%	-2.45%	-0.42%
1x total power pre-CTS	clock	-3.5%	10.1%	-23.9%	-73.8%	-0.26%	-2.80%	-0.53%
No LR sizer in post-CTS	route	-4.1%	5.5%	-75.5%	-82.8%	-0.28%	-2.79%	-0.75%
1x total_power pre-CTS	place	4.7%	0.6%			-0.10%	-0.27%	-0.01%
No LR sizer in post-CTS	clock	-0.9%	36.1%	-11.8%	-7.9%	-0.23%	-1.20%	-0.31%
	route	-17.5%	25.5%	-56.2%	-16.7%	-0.14%	-1.50%	-0.48%
2x total_power pre-CTS	place	7.0%	4.9%			-0.22%	-0.58%	-0.25%
No LR sizer in post-CTS	clock	-2.6%	8.9%	-23.4%	-8.0%	-0.23%	-1.54%	-0.34%
	route	-19.8%	-0.1%	-71.6%	-9.2%	-0.23%	-1.71%	-0.45%
total_power pre-CTS	place	3.9%	-3.8%			-0.10%	-0.31%	0.04%
total_power post-CTS	clock	-4.5%	1.8%	-16.1%	-55.3%	-0.84%	-2.29%	-1.03%
	route	-18.3%	2.7%	-56.4%	-59.8%	-0.62%	-2.54%	-0.94%
No LR sizer in pre-CTS	place	0.6%	3.6%			-0.01%	-0.17%	-0.07%
total_power post-CTS	clock	-4.2%	35.1%	17.2%	-54.5%	-0.88%	-3.32%	-1.03%
	route	-19.2%	23.0%	-71.2%	-65.9%	-0.57%	-2.32%	-0.59%

Ratio of number of better results to worse results for a metric, out of 67 designs.

Sizer Flow Experiment	Stage	WNS	TNS	WHS	THS	Area	Leakage	Tot. Pow.
leakage pre-CTS	place	35:21	37:23			38:29	41:18	39:28
No LR sizer in post-CTS	clock	34:23	27:32	16:14	22:9	36:30	40:17	40:27
	route	26:22	23:25	16:24	19:22	39:28	41:18	40:27
2x leakage pre-CTS	place	31:27	39:21			39:28	49:12	37:30
No LR sizer in post-CTS	clock	30:26	27:32	15:16	19:12	42:25	47:13	37:30
	route	21:26	23:24	21:16	18:21	39:28	42:17	36:31
leakage pre-CTS	place	35:22	35:25			37:30	40:19	37:30
leakage post-CTS	clock	29:27	24:34	17:14	21:11	47:20	52:8	44:23
	route	21:28	19:31	19:19	17:19	44:23	51:9	45:22
No LR sizer in pre-CTS	place	25:25	29:24			28:28	20:26	26:31
leakage post-CTS	clock	23:33	25:32	16:15	19:12	57:10	56:4	50:17
	route	23:24	22:25	19:17	22:15	52:15	55:6	44:23
1x leakage pre-CTS	place	32:25	35:24			48:19	46:16	46:21
1x total power pre-CTS	clock	30:28	27:32	17:15	21:11	47:19	44:15	45:22
No LR sizer in post-CTS	route	25:22	25:22	18:16	18:19	44:23	42:16	46:21
1x total_power pre-CTS	place	30:29	36:24			39:28	29:29	36:31
No LR sizer in post-CTS	clock	31:27	24:34	14:19	17:16	37:30	30:27	35:32
	route	23:25	21:28	20:15	21:16	38:28	32:26	36:31
2x total_power pre-CTS	place	32:28	35:28			48:19	39:22	40:26
No LR sizer in post-CTS	clock	29:31	28:32	17:14	18:14	39:28	38:21	37:30
	route	27:21	22:26	17:17	19:18	39:27	36:24	40:27
total_power pre-CTS	place	31:28	37:23			42:25	32:27	36:30
total_power post-CTS	clock	29:29	24:35	16:18	15:18	51:16	45:15	48:19
	route	25:25	19:32	15:20	17:19	47:20	44:16	43:24
No LR sizer in pre-CTS	place	27:22	26:28			28:27	26:22	27:29
total_power post-CTS	clock	35:22	29:29	13:17	18:12	62:5	50:8	50:17
	route	27:21	23:25	23:15	22:17	53:14	46:15	45:22

Full flow experiments enabling LR sizer – impact of removing TNS outlier

- The sum difference in TNS at the end of the clock and route stages is dominated by an outlier due to non-determinism in clock tree synthesis. (It is not due to enabling the LR sizer.)
- Baseline TNS of -8.5us for this outlier design worsened to -14.2 to -16.8us at the clock stage in some runs due to this.
- The table on the right shows the impact of omitting this outlier, namely better clock stage results. The route stage results are still dominated by another smaller outlier.

	Flow	Sum Difference of TNS	
Sizer Flow Experiment	Stage	With outlier	Removed outlier
1x leakage pre-CTS	place	-1.4%	-0.4%
No LR sizer in post-CTS	clock	52.5%	-0.5%
	route	44.6%	40.2%
2x leakage pre-CTS	place	2.2%	3.1%
No LR sizer in post-CTS	clock	27.9%	-7.7%
	route	23.6%	11.9%
1x leakage pre-CTS	place	-2.1%	-1.2%
1x leakage post-CTS	clock	48.1%	9.5%
	route	50.1%	49.4%
No LR sizer in pre-CTS	place	2.7%	3.7%
1x leakage post-CTS	clock	6.1%	-0.4%
	route	-2.3%	12.6%
1x leakage pre-CTS	place	-0.6%	0.1%
1x total power pre-CTS	clock	10.1%	-3.2%
No LR sizer in post-CTS	route	5.5%	9.3%
1x total_power pre-CTS	place	0.6%	1.6%
No LR sizer in post-CTS	clock	36.1%	-0.6%
	route	25.5%	6.0%
2x total_power pre-CTS	place	4.9%	5.8%
No LR sizer in post-CTS	clock	8.9%	3.0%
	route	-0.1%	10.1%
1x total_power pre-CTS	place	-3.8%	-3.0%
1x total_power post-CTS	clock	1.8%	4.8%
	route	2.7%	17.6%
No LR sizer in pre-CTS	place	3.6%	4.3%
1x total_power post-CTS	clock	35.1%	-5.1%
	route	23.0%	3.6%