Accelerated design convergence using automatic recipe tuning through Intel enabled AI-ML PnR framework

Rohit Murarka, SoC Design Engineer, Intel
Raghavendra Vasappanavara, Architect, Intel
Jagadeesh Boddu, SoC Design Engineer, Intel
Narendra Nimmagadda, Engineering Manager, Intel

Motivation

As we enter Armstrong Era of fabrication, innovative ways to converge complex designs with faster TAT and best-in-class PPA & QoR is becoming significant.

Manual design convergence efforts have its own challenges:

- Trying out various design exploration experiments to find out best recipes.
- Manual Analysis done across multiple RTL2GDS runs.
- Time consuming and unpredictable timelines.
- Recommended recipes can still be error-prone and non-optimal.

The problem is exacerbated for early netlist designs:

- Full RTL2GDS Design collaterals are not stable.
- Added challenge of navigating through collaterals stability issues while finding out best recipes.

Main Idea

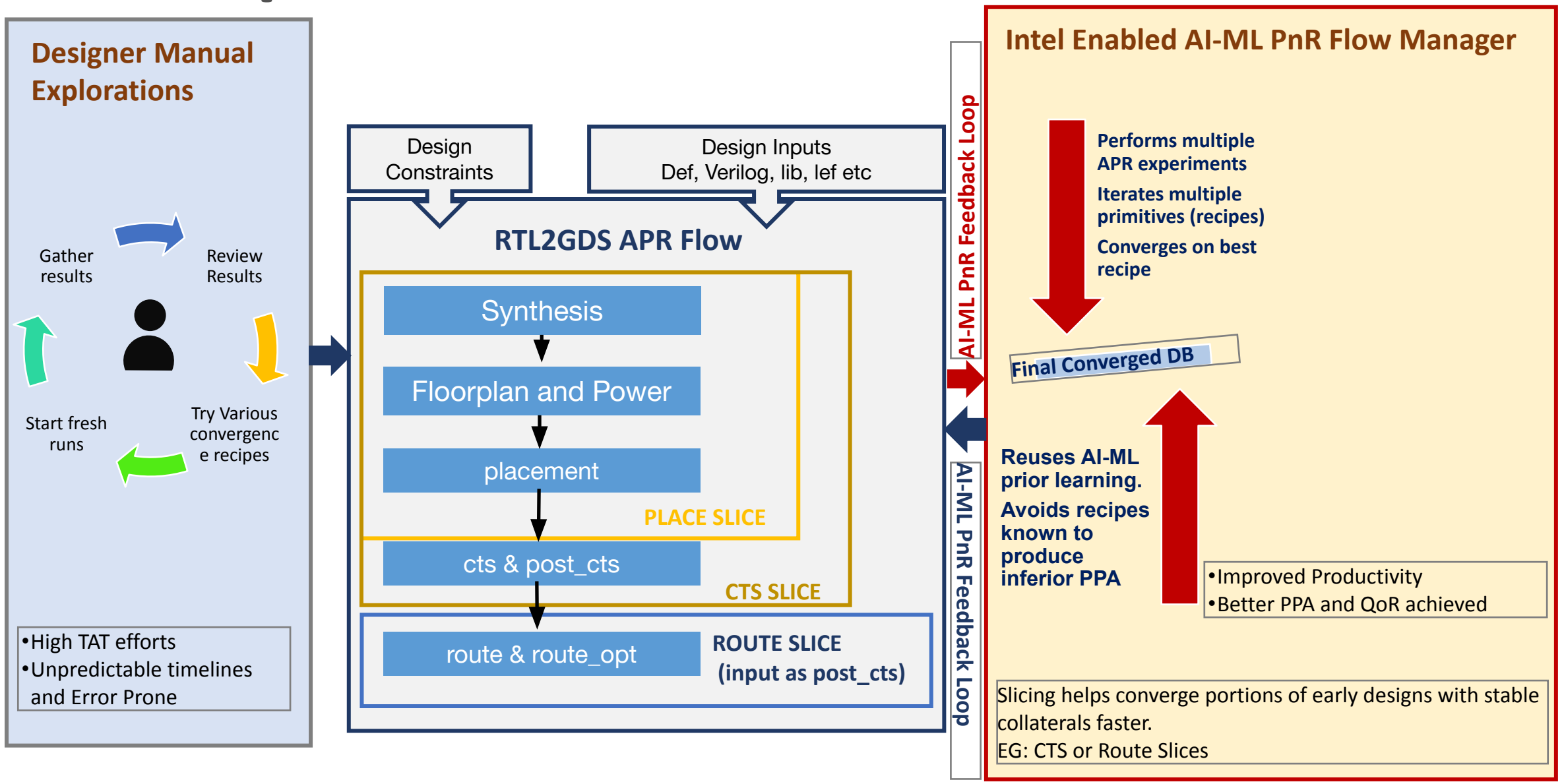
Propose Intel enabled AI-ML PnR framework:

- Helps designers find out best PPA convergence recipe in automated way and timebound manner.
- Takes designer guard bands as input other than regular design collaterals.
- Ensures recommended recipe meets general design requirements.

For early Netlist design with unstable collaterals for full RTL2GDS Flow:

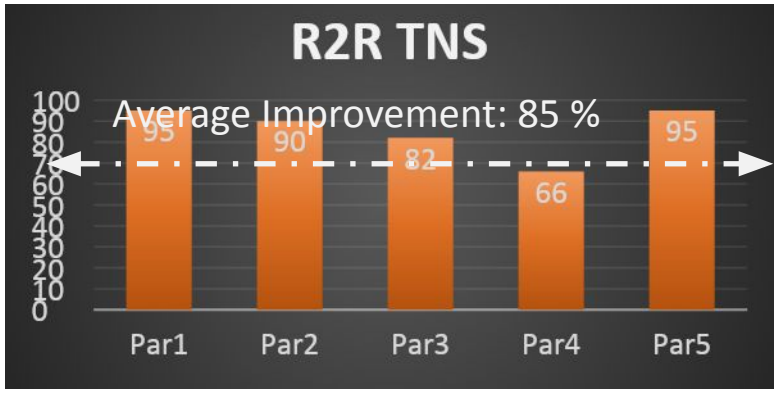
- AI-ML recipe recommender can be sliced only for portions of PnR flow stages where usable collaterals are available.
  - EG: Place Slice, CTS Slice, Route Slice
- Helps improve feedback loop and best suitable recipes with faster TAT.
- Enables faster design maturity with timely feedback on collateral quality.

Idea Implementation Detail

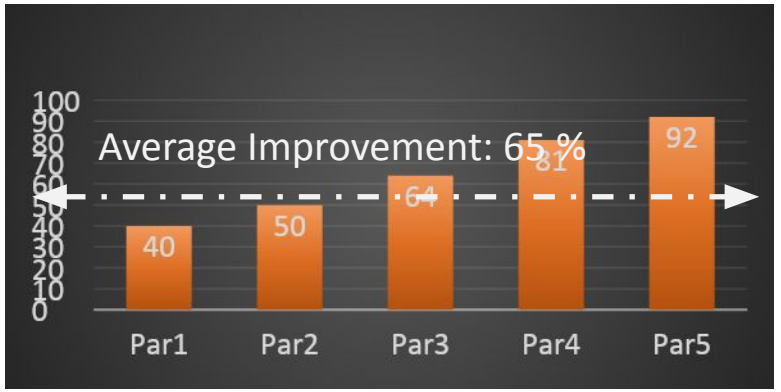


Results

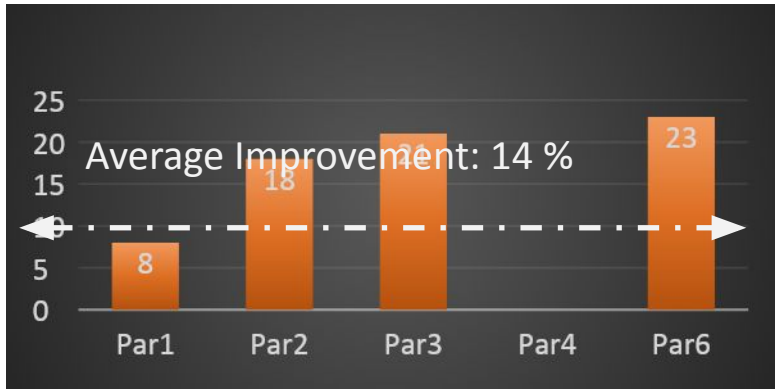
Goal: TNS Improvement



Goal: WNS Improvement



Goal: Leakage Improvement



Intel enabled AI-ML PnR recipes showcasing QoR (TNS, WNS, Leakage) Improvements over and above Manual RTL2GDS runs @route_opt

Major Takeaways from Intel Enabled AI-ML PnR flow

- Optimized flows generated better timing and power results for every block, compared to default Intel flow
- Significant power benefit for complete SoC
- Automation enabled engineering team to tune the flow for every block
- SoC tapeout complete using AI-ML optimized flows

Summary

Intel enabled AI-ML PnR flow deployed on Intel Design successfully.

- Showcased faster design convergence with Improved QoR.

- **Challenges**
  - Requires humongous Disk Space, Compute availability commits from projects since it does multiple explorations in parallel.
  - AI-ML PnR flow found to be extremely sensitive to Design, collateral or flow quality and stability.
- **Look Ahead**
  - Dirty data handling to enable better exploration for early as well as intermediate design quality.
  - Enable AI-ML PnR Cloud based Design convergence to help offload disk and compute challenges.
  - Enable various design explorations using AI-ML PnR. Few design explorations areas of interest:
    - Block sizing
    - Macro Placements
    - Power-Grid structure
    - Frequency Tuning
    - Port Placement