IWLS 2026 Programming Contest

Alan Mishchenko, UC Berkeley Attila Jurecska, Siemens EDA

The IWLS Programming Contest this year continues the logic-synthesis competitions held at IWLS in recent years. In contrast to previous years, the objective of this contest is not to find a single "best" solution per benchmark. Instead, the goal is to explore the tradeoff between two discrete optimization objectives:

- Area = the number of internal two-input AND nodes in the AIG (node count).
- Delay = the number of logic levels in the AIG (level count).

Because both objectives are discrete, each benchmark has (in principle) many meaningful solutions, ranging from a small area with larger delay to a small delay with larger area. The goal of this year's contest is to produce a high-quality Pareto curve (Pareto set) of (delay, area) tradeoff points for each benchmark, and to compare the quality of synthesis results produced by the participants, based on the quality of these tradeoff curves.

Track and circuit representation: There is one track: synthesizing traditional And-Inverter Graphs (AIGs). Each solution is an AIG written in the standard binary AIGER format (https://fmv.jku.at/aiger/). As in previous contests, all submitted AIGs must be functionally equivalent to the given truth table.

Benchmarks: 100 new practical benchmarks, with file names ex200.truth through ex299.truth, were generated for this contest (with possibly a few duplicates from the previous years). The high-level descriptions are not disclosed because this is a logic synthesis competition and the participants are encouraged to develop optimization methods that are universally applicable, rather than tailored to specific known circuits.

Submissions: Each team submits one zip file whose name includes the team's affiliation. Please do not submit many zip files and do not create subdirectories inside the archive, only optimized AIGs with the proper file naming in one zip file. For each input truth table "ex2NN.truth", a team may submit any number of AIG files corresponding to different (delay, area) tradeoff points. The required file naming convention is ex2NN_LLL.aig, where LLL is the level count of a testcase with leading zeros up to three decimal digits. For example, some of the submitted files may look as follows: ex200_030.aig, ex200_031.aig, ex200_032.aig, etc, meaning that this is the first testcase (ex200.truth) optimized as a series of AIGs having 30, 31, 32, etc, logic levels.

Important: Teams are strongly encouraged to submit only Pareto points (see below). If a team submits redundant points, compared to other points submitted by the same team, they will be automatically removed and not considered for evaluation.

Functional correctness: Each submitted AIG is checked for functional equivalence against the corresponding truth table using ABC (as in previous years). For example, an AIG for "ex200.truth" can be checked with "read_truth -xf ex200.truth; cec -n ex200_030.aig" or with "read_truth -xf ex200.truth; st; &get; &cec -t ex200_030.aig". Functionally incorrect AIGs will be discarded.

Pareto points: Each submitted correct AIG corresponds to one point (delay, area) in the solution space. A point is considered "non-Pareto" (dominated) and will be removed if the same team has another submitted point that is at least as good in both objectives and strictly better in one of them. In other words, a point is removed if the team's submission already includes a point that has better area and better delay, the same area and better delay, or the same delay and better area.

The "virtual best solver": For each benchmark, after processing all teams' submissions, as described above, the organizers will form the union of all teams' points and again remove dominated (non-Pareto) points. The remaining global Pareto set is called the "virtual best solver" for this benchmark. Intuitively, it is the best tradeoff curve that could be obtained by taking, for each tradeoff region, the best point of any participant.

How participants are compared to the virtual best solver: The comparison is performed "delay by delay" using the discrete delay values appearing in the virtual best solver's Pareto set:

- Consider one Pareto point of the virtual best solver with a specific level count L. This point represents the best known area achieved by any participant at that delay.

- For the participant being scored, we find the participant's best available area among their own Pareto points whose level count is not larger than L (i.e., any of their points that are at least as fast as L). If the participant has no such point, the participant receives 0 points for this delay value.

- Otherwise, the participant receives an integer number of points based on how close their area is to the virtual best area at this delay. As in prior IWLS contests, this is computed as the integer part of:

  100 × (virtual-best area) / (participant's area)

  Thus, matching the virtual best area gives 100 points at that delay value; larger areas give proportionally fewer points.

Benchmark score and final score: For each benchmark, the participant's score is the average of the points obtained over all delay values (Pareto points) in the virtual best solver's Pareto set for that benchmark. The participant's final contest score is the floating-point average of the benchmark scores over all 100 benchmarks. The winner is the team with the highest final score.

Notes:

- Submitting more (correct) points is only helpful if they improve the participant's tradeoff curve. Non-Pareto points will not be considered.
- Participants can use ABC (https://github.com/berkeley-abc/abc) to validate correctness and to measure area and delay while developing their methods, as in previous years.