

AxLS: An Open-Source Framework for Netlist Transformation Approximate Logic Synthesis

Jorge Castro-Godínez^{1,2}, Humberto Barrantes-García², Muhammad Shafique³, Jörg Henkel¹

¹Karlsruhe Institute of Technology, ²Instituto Tecnológico de Costa Rica, ³New York University Abu Dhabi

CES – Chair for Embedded Systems



Approximate Computing

- Computational *quality* (accuracy of results) vs. computational *effort* (execution time, area, power, or energy).



Input image



PDP = 1.00 / PSNR = inf dB



PDP = 0.69 / PSNR = 29.8 dB



PDP = 0.62 / PSNR = 22.4 dB



PDP = 0.61 / PSNR = 19.52 dB

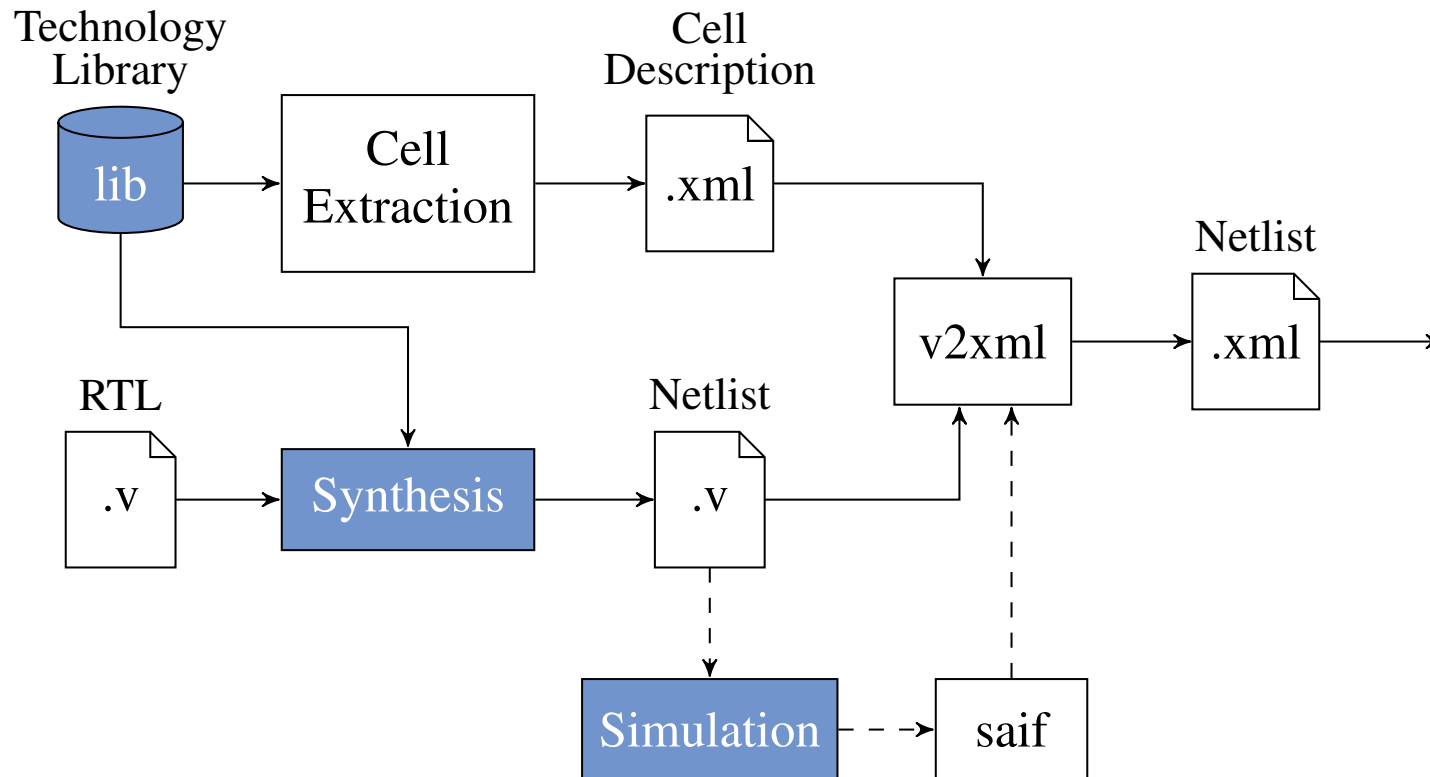
Motivation

- Approximate Logic Synthesis (ALS)
 - Generate approximate circuits from accurate implementations.

- Functional simplification:
 - Netlist transformation.
 - Boolean re-writing.
 - Approximate High-Level Synthesis.

- Missing an open-source framework for netlist transformation-based.

AxLS – Netlist Generation

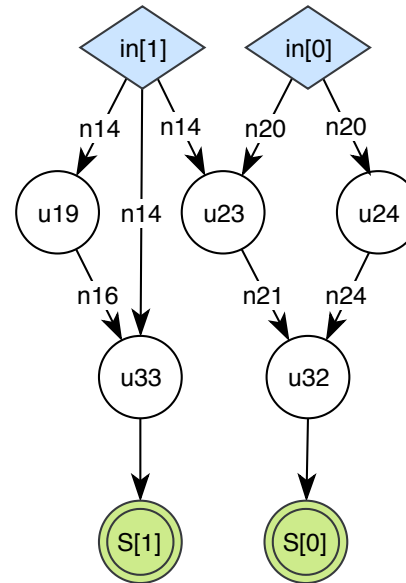


AxLS – XML-based Netlist

```

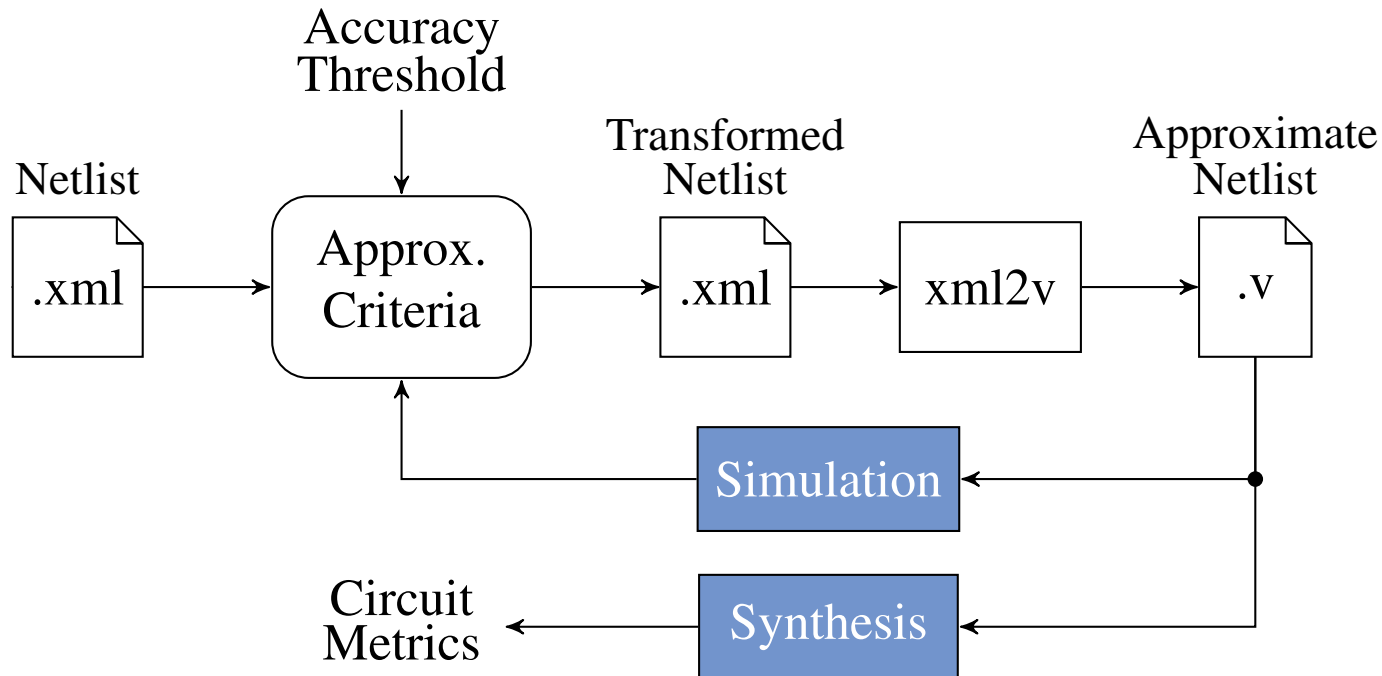
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <node name="INV_X2" var="u19">
    <input name="A1" wire="n14" />
    <output name="ZN" wire="n16" />
  </node>
  <node name="OAI21_X1" var="u23">
    <input name="A1" wire="n14" />
    <input name="A2" wire="n20" />
    <output name="ZN" wire="n21" />
  </node>
  <node name="NAND2_X2" var="u33">
    <input name="A1" wire="n16" />
    <input name="A2" wire="n14" />
    <output name="ZN" wire="S[1]" />
  </node>
  [ ... ]
<circuitinputs>
  <input var="in[0]" />
  <input var="in[1]" />
</circuitinputs>
<circuitoutputs>
  <output var="S[0]" />
  <output var="S[1]" />
</circuitoutputs>
</root>

```



DAG of Netlist Description

XML Netlist Description



AxLS – Evaluation

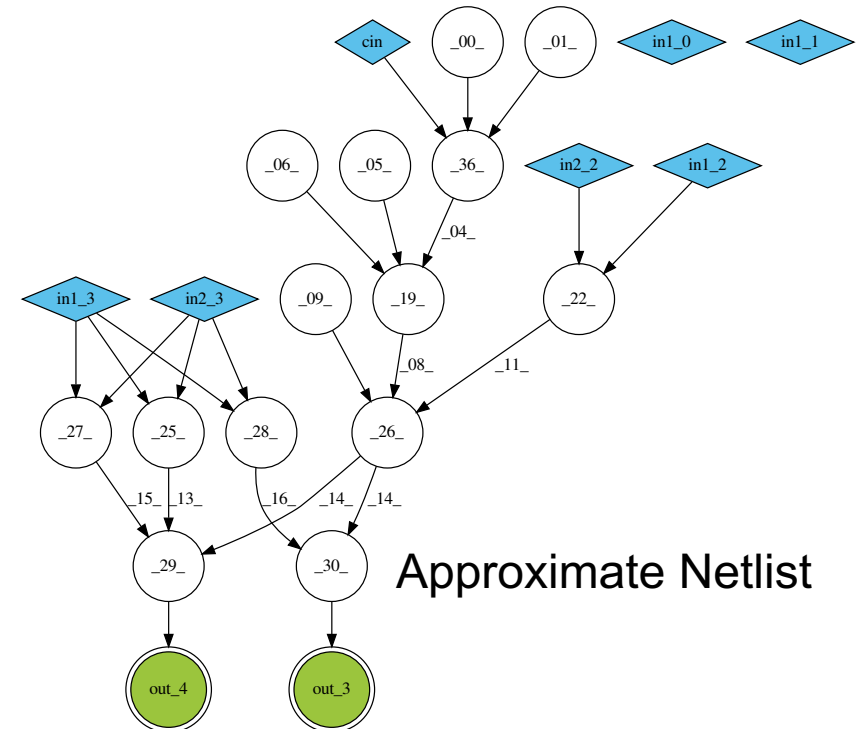
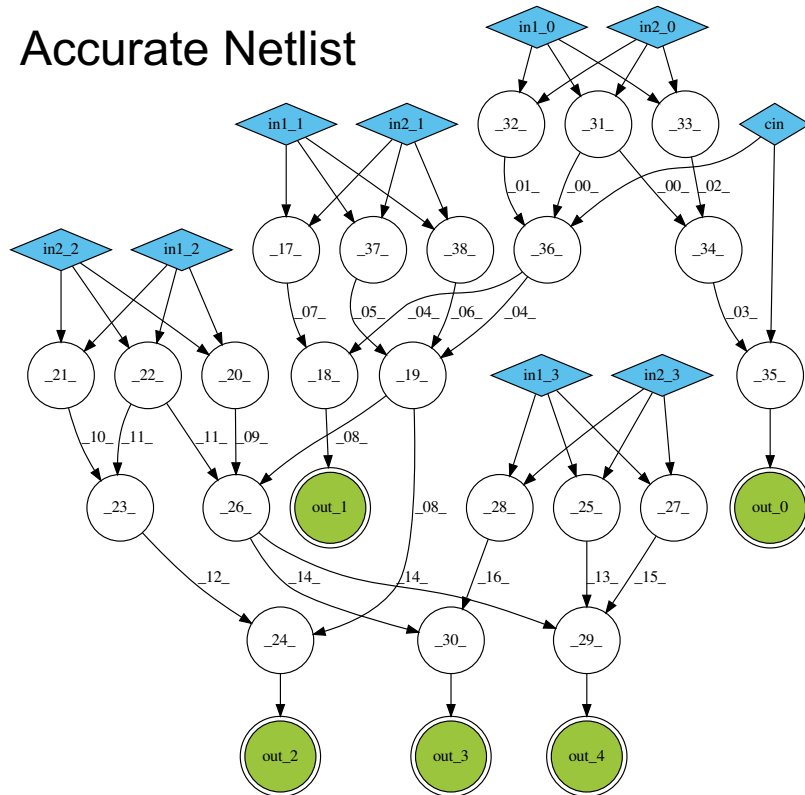
- AxLS implemented using Python language.
- External tools:
 - Yosys for circuit synthesis.
 - Icarus Verilog for netlist simulation.
- Arithmetic circuits, particularly standard adders.
- NanGate 15 nm technology library.

AxLS – Evaluation

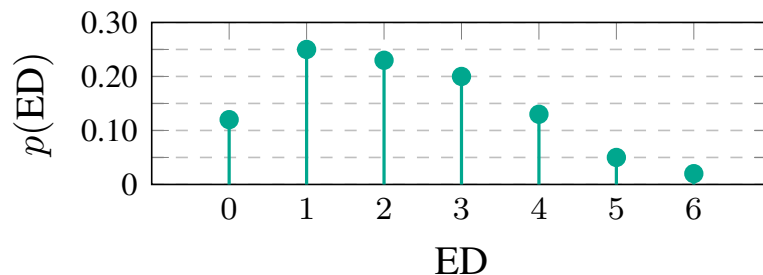
- Example of pruning technique, as Approximation Criteria:
 - Primary input constant.
 - All dependencies explored.
 - Primary output constant.
 - Explore nodes affecting such output.
- Nodes removed, replaced with 0 value.
- Starting from LSB.

AxLS – 4-bit RCA

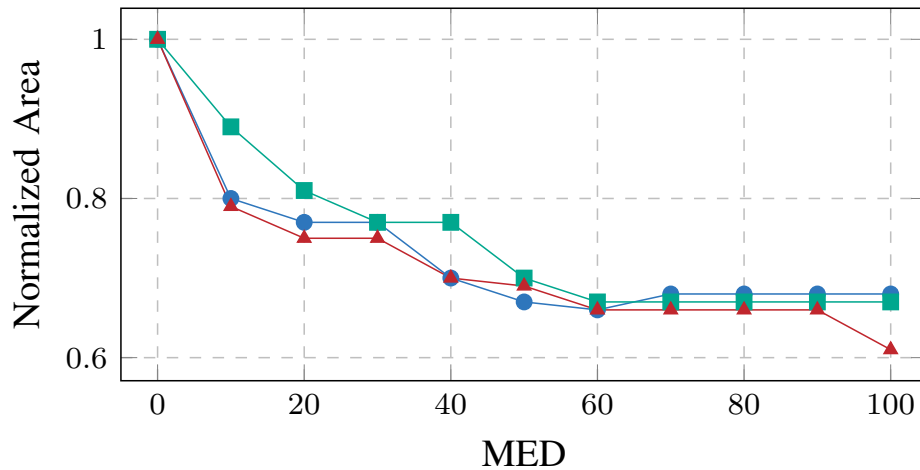
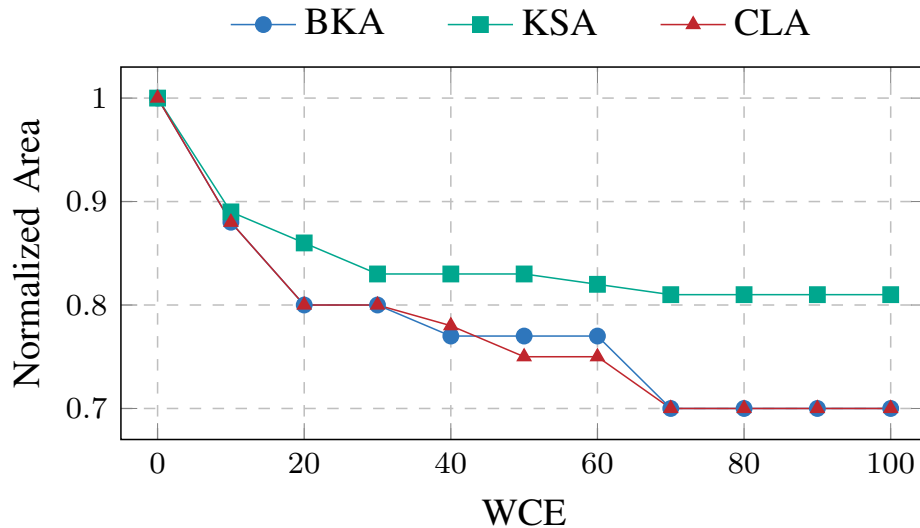
Accurate Netlist



Approximate Netlist



AxLS – Evaluation



Summary

- AxLS is an open-source framework for ALS techniques based on netlist transformation.
- Available at <https://github.com/ECASLab/AxLS>
- Example of pruning techniques that can be applied, for different accuracy metrics, and tested with standard adders.
- As future work, ML-based techniques for error estimation of gate pruning.

The End



Schloss Karlsruhe (*Karlsruhe Palace*)

Thanks for your watching!

