Develop framework for explainable DSE of deep learning accelerators that reasons about underlying inefficiencies in designs, achieve efficient designs, takes short time, and can work with several domains.

**Background**
- Effective design space exploration (DSE) requires achieving efficient solutions satisfying constraints under practical exploration budgets.
- Deep learning accelerator design space can be vast.
- Recent industrial/academic approaches use black-box DSEs.
- Evolutionary algorithms.
- ML-based approaches.
- Cost models take milliseconds—Practical explorations can afford only 1000s of iterations.

**Table 1: Design Space for Edge DNN Accelerators**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1 buffer (b)</td>
<td>64, 128, 256, 512</td>
</tr>
<tr>
<td>L2 buffer (B)</td>
<td>64, 128, 256, 512</td>
</tr>
<tr>
<td>OS/WS bandwidth (1004, 2484, 4886, 1600, 3426)</td>
<td>(Mbit/s)</td>
</tr>
<tr>
<td>NCH bottleneck</td>
<td>16% in (1, 16)</td>
</tr>
<tr>
<td>Physical unit (ps)</td>
<td>60% in (10, 10)</td>
</tr>
<tr>
<td>Virtual unit (ps)</td>
<td>3% in (10, 10)</td>
</tr>
</tbody>
</table>

**Problem Statement**
- Black-box DSEs are non-explainable.
  - Can’t reason about why sampled configurations incur higher costs and how to change parameters to address underlying execution inefficiencies.
  - Process single cost value for a DNN vs. per-layer costs and do not leverage information about domain-specific bottlenecks.
  - They require excessive trials (thousands), which leads to:
    - Low Efficiency of obtained solutions (several-fold).
    - Low Feasibility (most of acquired solutions do not meet constraints).
    - Cannot do Runtime/Practical DSE (Takes days—weeks).

**DSE Using Bottleneck Analysis & API for Expressing Domain-Specific Bottleneck Models**
- Through API, architects/tools can specify bottleneck graph of target costs and appropriate scaling of parameters.
- Workflow example for solution acquisitions addressing multiple bottlenecks and constraints-budget awareness.

**Motivation**
- Explainable DSEs majorly improves feasible solutions (37% vs. 21% for non-explainable DSEs).
- Including software design space in the exploration enables 4.24x better solutions and tightly coupled codesign (search time increases from 21 to 64 minutes; from 16 to 21 minutes for most DNNs).

**Results**
- **Consistent/Quick Objective Reduction**
  - Constraints-budget-aware DSE majorly explores feasible solutions (37% vs. 21% for non-explainable DSEs).
  - Including software design space in the exploration enables 4.24x better solutions and tightly coupled codesign (search time increases from 21 to 64 minutes; from 16 to 21 minutes for most DNNs).

**Conclusions**
- Explainability can be enabled by domain-awareness and bottleneck analysis.
- Prior works lack formal specification of how to express bottleneck analysis in a generic manner for an effectual design space exploration.

**Future Directions**
- Automate bottleneck model generation for a variety of domain-specific architectures.
- Introducing bottleneck mitigations in ML-based DSE.
- Overcome local-optima-convergence due to greediness for mitigating primary bottlenecks.

**Publications**

**Website and Related Material:**