SEPO: Using symbolic states to find RISC-V optimizations

Symbolic Memory

Algorithms:

M := []

load(e):
if e is concrete:
  if e not in M:
    M[e] := new_symbol()
    return M[e]
else:
  if ∃ e’ in M s.t. e ≡ e’:
    return M[e ‘]
  M[e] := new_symbol()
  return M[e]

store(e, v):
for e’ in M:
  if is_sat(e = e ‘):
    M[e] = e’
  M[e] = v

Future Work:
- Synthesize equivalent programs using symbolic states
- Incorporate value liveness using CFA
- Handle control flow