

Neural Functional Programming

John K. Feser, Marc Brockschmidt, Alexander L. Gaunt, Daniel Tarlow

Inductive Program Synthesis

Programming by example

Given: input-output example (or several)

Inputs: [3, 1, 7, 4, 2], 2

Output: [5, 3, 9, 6, 4]

Task: find program consistent with IO example(s)

```
let r0 = Input() in
let r1 = Input() in
let r2 = map (λ elem → elem + r1) r0 in
r2
```

Idea: Use programs as hypothesis space, and use gradient descent to search for programs.

Question: What programs are suited for this?

Differentiable Interpreters

Describe structure of an interpreter, leave source code as parameters to be inferred.

Insert NAMPI TerpreT poster picture here

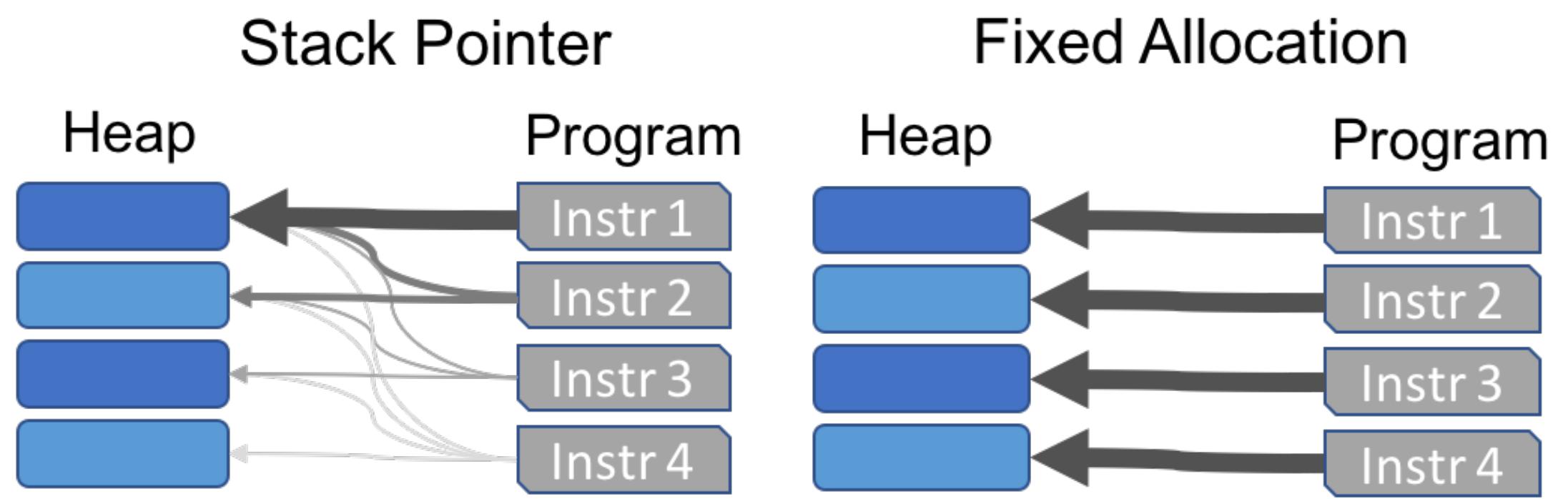
Idea C: Use Structured Control Flow

Interpreter supports if-then-else, foreach, map, zipWith, fold instead of gotos.

```
pre1:  $\boxed{\text{out}}$   $\leftarrow$   $\boxed{\text{instr}}$   $\boxed{\text{arg1}}$   $\boxed{\text{arg2}}$   $\boxed{\text{cond}}$ 
pre2: ...
foreach ele in  $\boxed{\text{reg}}$ 
loop1:  $\boxed{\text{out}}$   $\leftarrow$   $\boxed{\text{instr}}$   $\boxed{\text{arg1}}$   $\boxed{\text{arg2}}$   $\boxed{\text{cond}}$ 
loop2: ...
suf1:  $\boxed{\text{out}}$   $\leftarrow$   $\boxed{\text{instr}}$   $\boxed{\text{arg1}}$   $\boxed{\text{arg2}}$   $\boxed{\text{cond}}$ 
suf2: ...
```

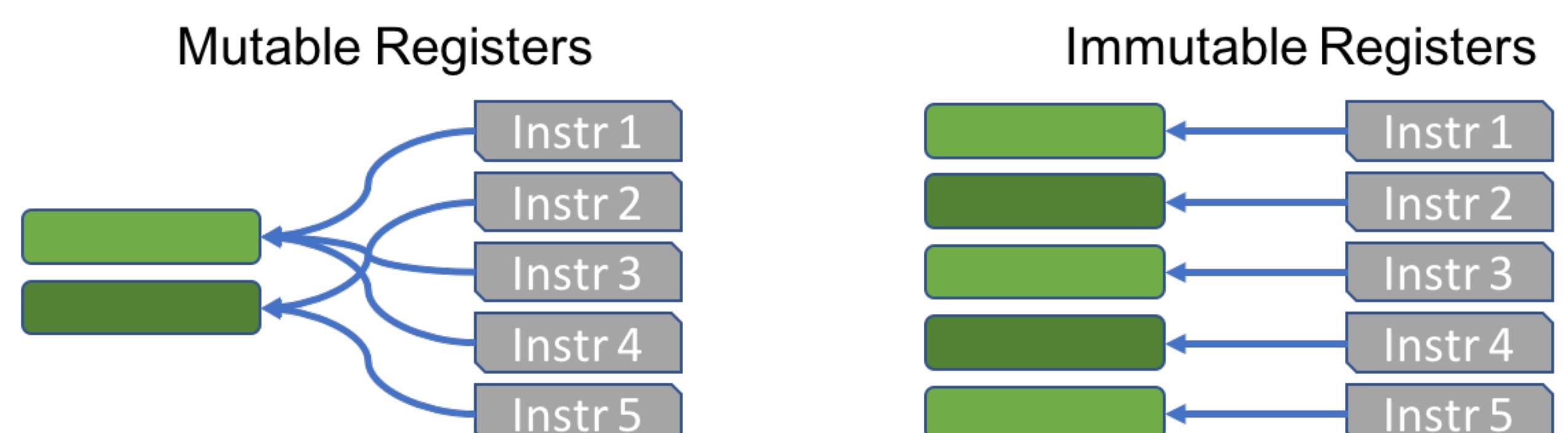
Idea F: Fixed Memory Allocation

Interpreter defines fixed scheme where to place newly computed heap values.



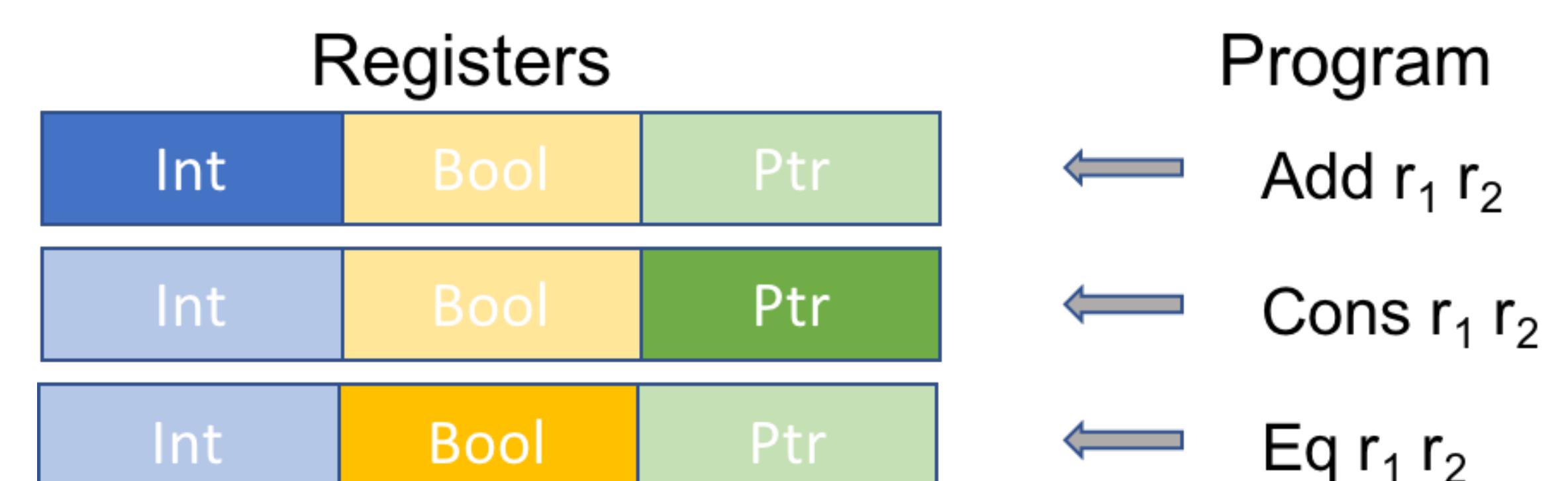
Idea I: Immutable Data

Interpreter defines output for every instruction.



Idea T: Leverage Types

Store values of different types separately.



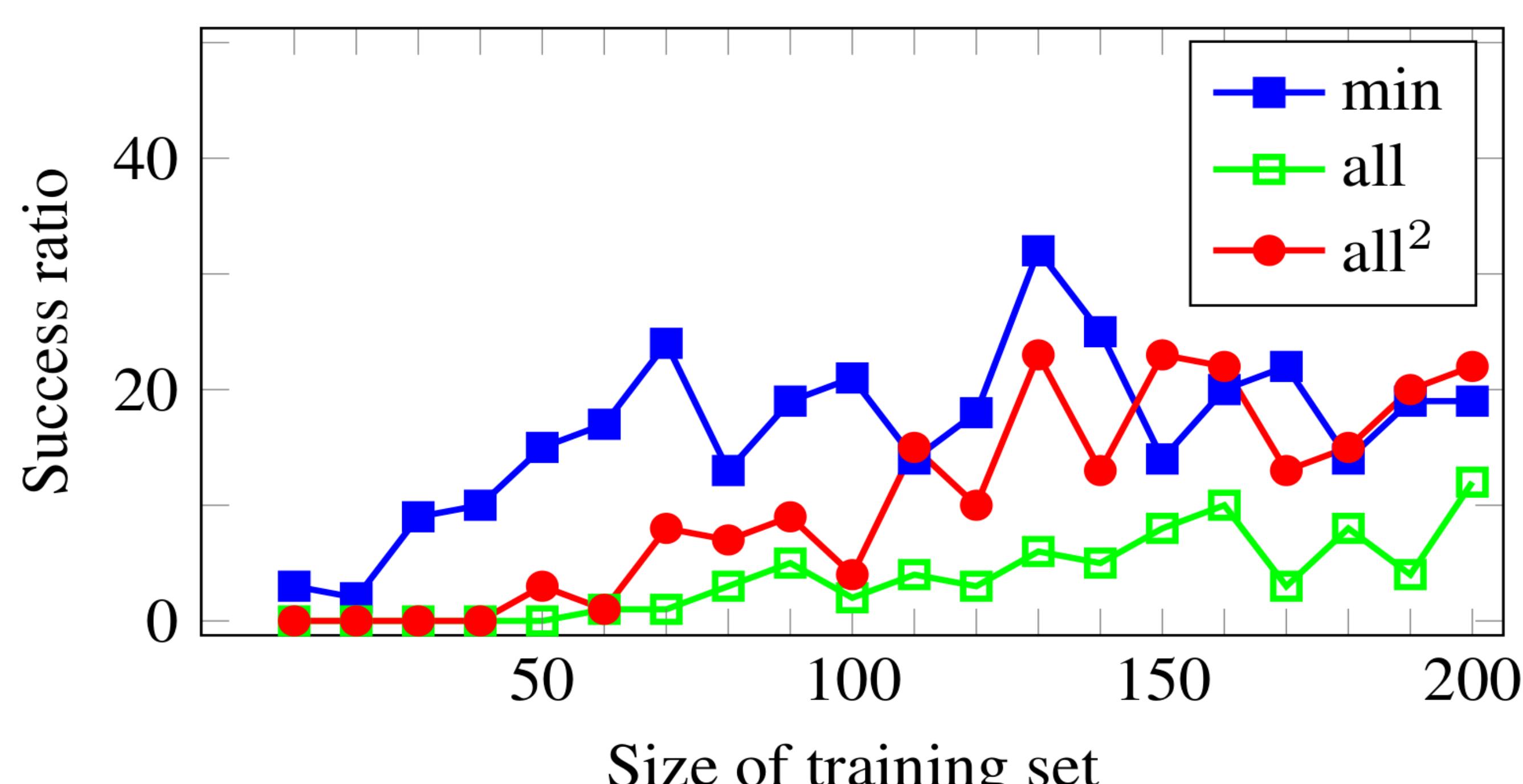
Each instruction writes into a typed register slot. No mixing of differently typed data.

Experiments

Study ratio of runs from random initialization converging to strongly generalizing solution for task on 5 I/O examples.

Task	C+T+I	C+T	C+I	C	λ^2
len	98.67	96.33	0.67	0.33	100.00
rev	18.00	10.33	2.67	8.33	100.00
sum	38.00	38.33	1.00	0.00	100.00
allGtK	0.00	0.00	0.00	0.33	100.00
exGtK	3.00	1.00	0.67	0.00	100.00
findLastIdx	0.33	0.00	0.00	0.00	0.00
getIdx	1.00	0.00	0.00	0.00	0.00
last2	0.00	8.00	0.00	2.00	0.00
mapAddK	100.00	98.00	100.00	95.67	100.00
mapInc	99.67	98.00	99.33	97.00	100.00
max	2.33	5.67	0.00	0.00	100.00
pairwiseSum	43.33	32.33	43.67	33.67	100.00
revMapInc	0.00	0.67	0.00	0.00	100.00

Baseline: Neural Random Access Machine with different number of instructions.



Related Work

- **Neural Random-Access Machines** (Kurach et al., 2016)
- **Neural Programmer-Interpreters** (Reed and de Freitas, 2016)
- **Hybrid computing using a neural network with dynamic external memory.** (Graves et al., 2016)
- **Programming with a Differentiable Forth Interpreter** (Riedel et al., 2017)