

可微分抽象机混合编程系统

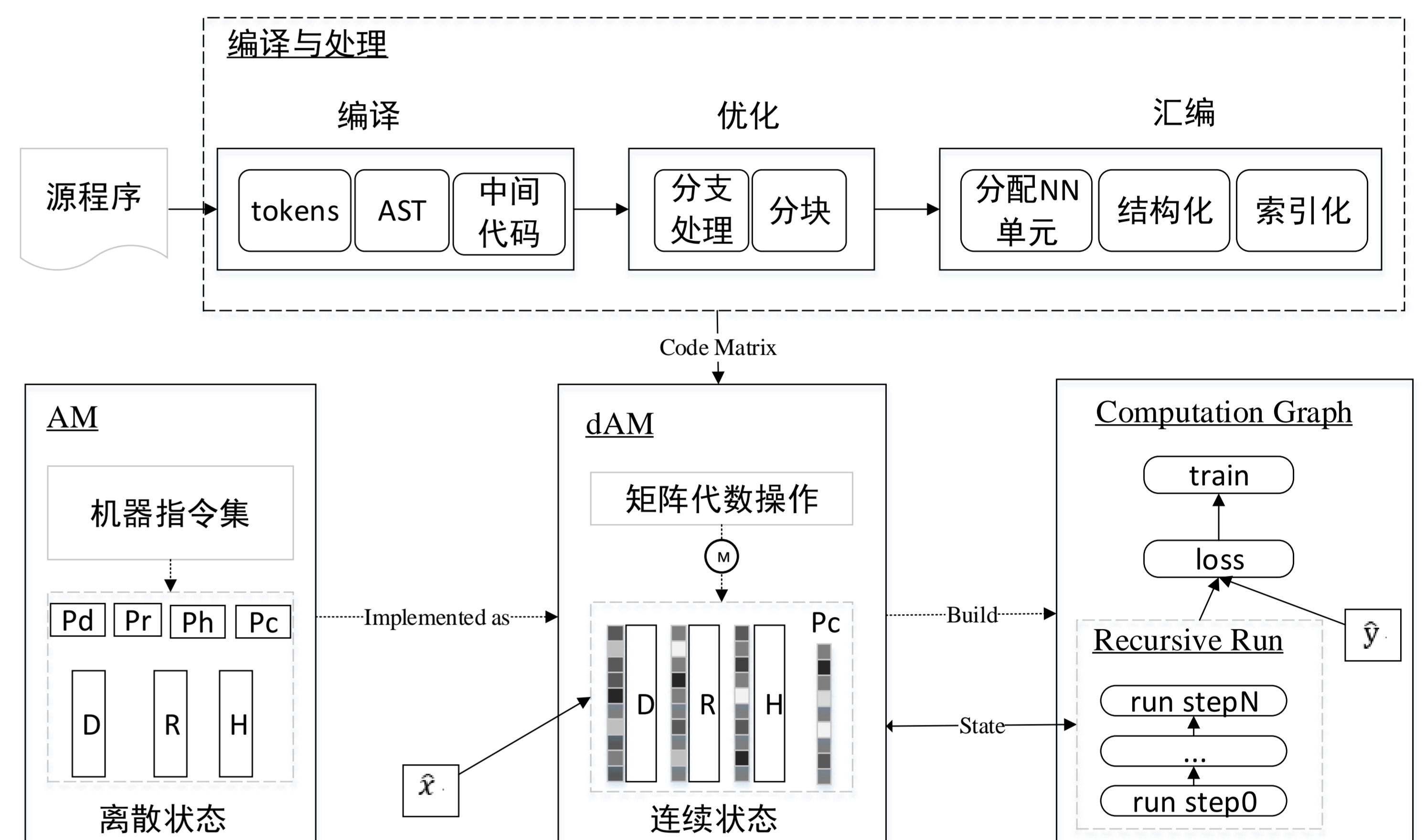
周鹏 武延军 赵琛

可微分抽象机混合编程系统, 软件学报, 2018 [录用]

联系方式: 周鹏 zhoupeng@iscas.ac.cn

背景:

本工作是软件所智能软件研究中心在自动编程上的研究成果之一。自动化编程是智能软件核心挑战之一, 程序执行轨迹和输入输出样例学习是典型研究方法, 这些方法无法弥合常规程序元素与神经网络组件差异, 不能吸收经验信息输入、缺乏编程控制。该工作给出一种无缝结合高级编程语言与神经网络组件的混合编程模型。优点是可使用过程化的高级编程语言元素和神经网络组件元素混合开发应用程序, 发挥并整合高级过程化编程和神经网络可训练学习编程各自的优势, 为基于神经网络的程序自动生成模型提供经验知识的输入接口。

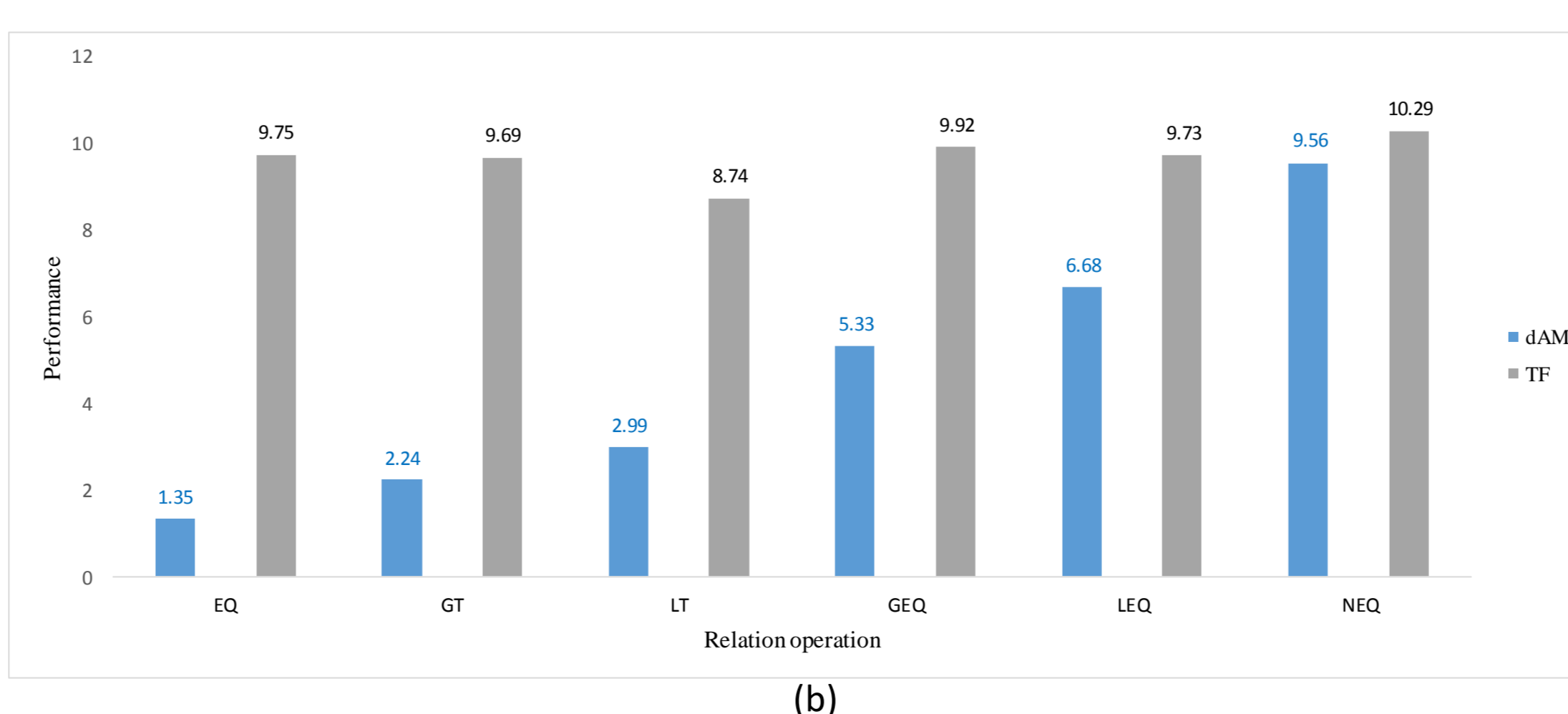
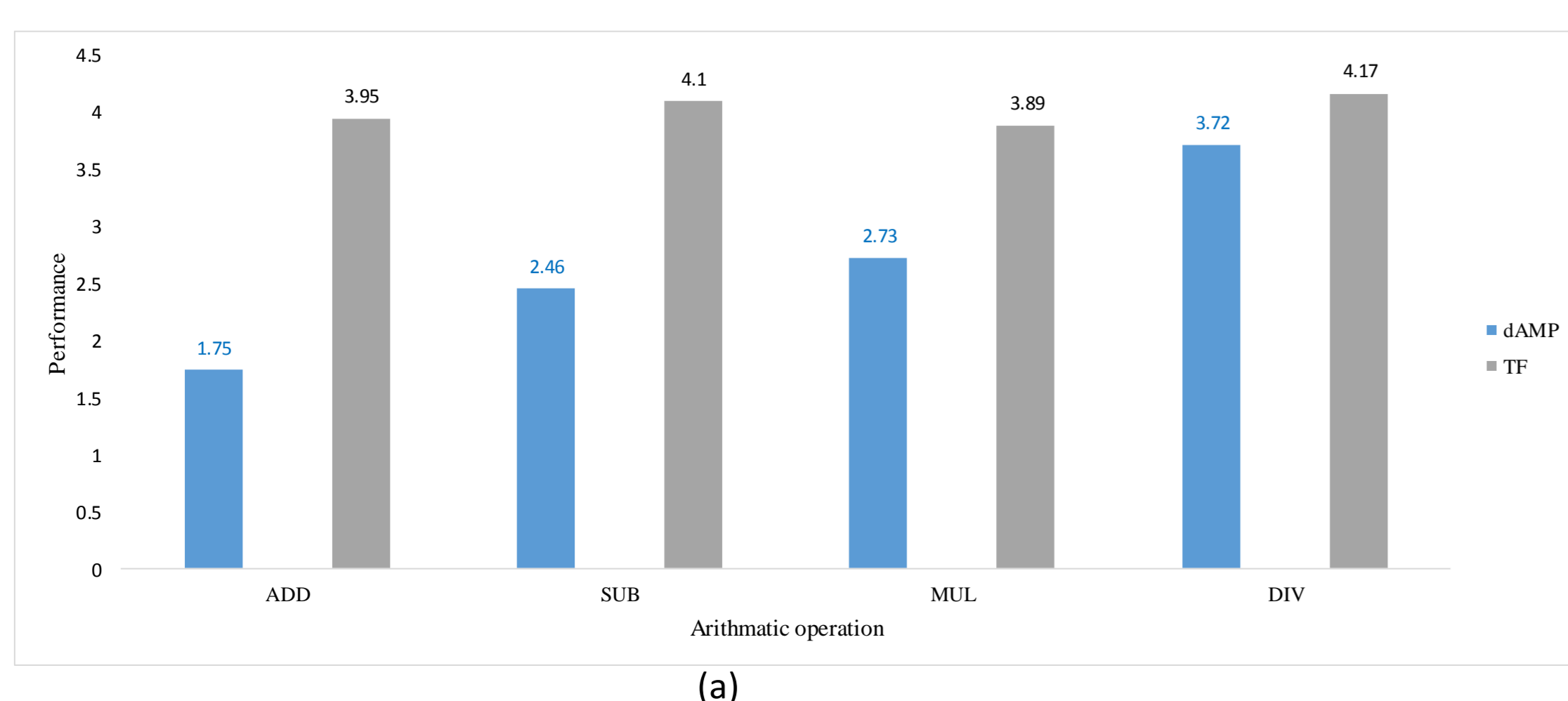
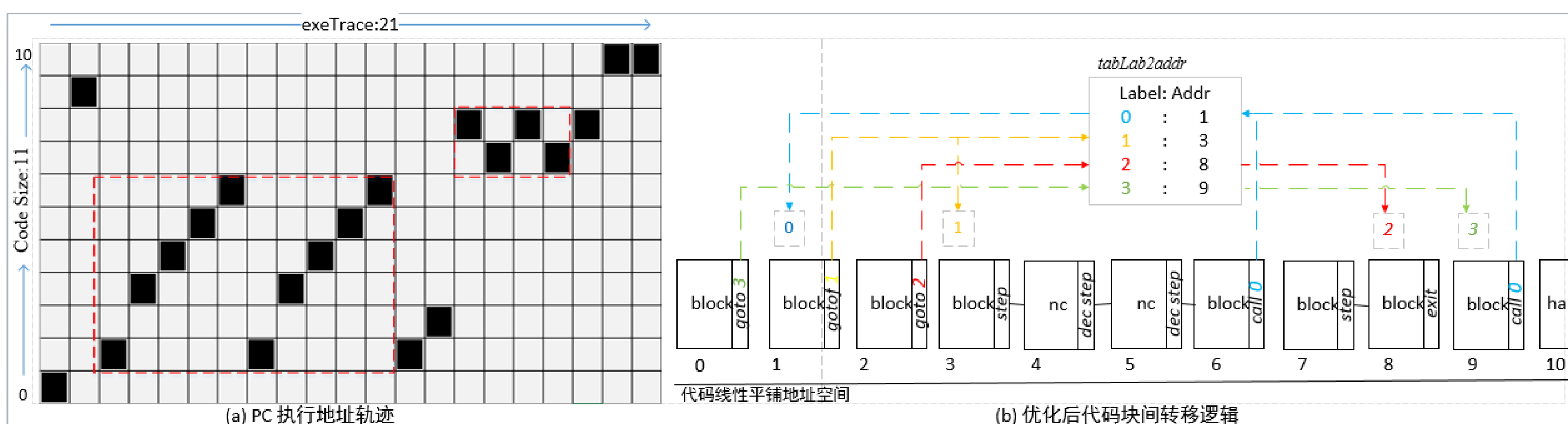


创新点:

- 提出在抽象层研究程序自动生成问题的一种思路, 并给出该思路的一个完整验证系统dAMP。
- 给出了一种支持经验输入、编程语言跟神经网络组件元素相结合的混合编程模型, 并给出了其设计实现的关键技术。

实用性:

- 可描述广泛复杂程序: 模型支持顺序块、条件分支、函数调用、递归等常规复杂程序构造和神经网络组件, 具有很好的编程表达能力。
- 非确定性编程: 不完全明确但有数据可依赖的问题编程, 支持以传统过程化编程方式在确定性框架下引入非确定性编程支持。
- 非完整编程: 整合神经网络和程序员各自的优势, 降低编程难度或提高编程效率。



```

1 : multiAdd (-)
  p_global @ n_global @ = if
  -exit \return
  else
  2 c_global @ \D-2
    rt_global p_global @ CELLS + @ \D-1
    lt_global p_global @ CELLS + @ \D0
    2DUP >R >R DUP >R \在R中生成a1,b1,c的副本
    ++ \加数、进位、被加数和和 sum=b1+c, sum
    R> SWAP R> SWAP R> SWAP
    R> @R 10 / \sum移动到R做副本,再复制回D
    R> 10 MOD \从R移动sum副本到D,本位和sum
    s_global ! \s=sum,需要保存或转移的栈上结果
    c_global ! \c=carry,需要保存或转移的栈上结果
    drop drop drop \drop D-2 D-1 D0,不需要保存
    s_global @ sum_global p_global @ CELLS + !
    p_global @ 1 + p_global ! \p++
    multiAdd
  2 then
  3 multiAdd
    carry @ . sum @ . \print the sum
  
```