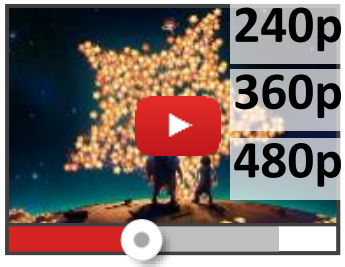
清華大學
Tsinghua University

智能网络系统的实用性与可解释性研究

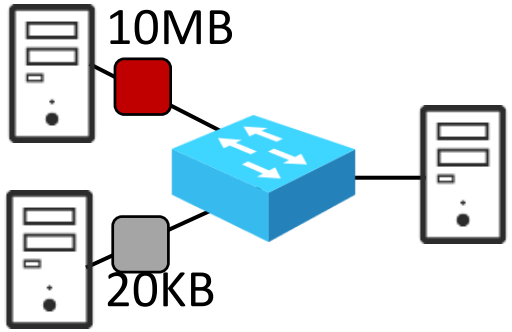
综合论文训练答辩

姓　　名：孟子立
指导教师：毕　军 教授

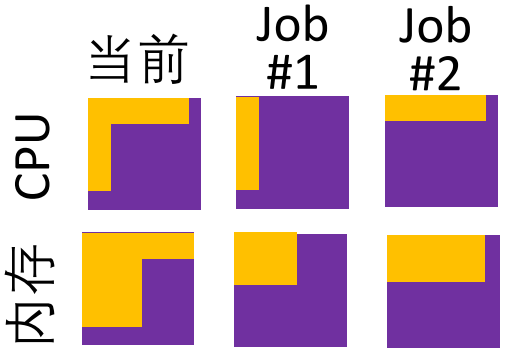
- 深度强化学习(Deep Reinforcement Learning, DRL)已经被广泛应用在网络系统中:
  - 由于其表达能力强、训练方便，DRL在许多网络系统中取得了很好的效果：



视频传输
(Pensieve[SIGCOMM'17])



流量工程
(AuTO[SIGCOMM'18])



任务调度
(Decima[SIGCOMM'19])

实际部署中存在的问题

决策延迟长

- 网络系统对延迟十分敏感：在10Gbps下，1毫秒的延迟会导致1.25MB的数据包被阻塞，使得缓冲区溢出。
- 在数据中心网络中，大部分数据流的流完成时间在毫秒级。
- 将基于DRL的数据中心流调度系统AuTO[SIGCOMM'18]实际部署时会带来100毫秒的前向传播延时，在这一段时间内95%的数据流已经完成。

实际部署中存在的问题

决策延迟长　资源消耗高

- 网络系统所部署的设备一般资源高度受限（如：交换机、移动端设备等）。
- 例如，采用Pensieve[SIGCOMM'17]时需要在加载视频前下载神经网络模型，可能需要10秒以上的延迟。
- 交换机等设备上甚至根本无法运行神经网络。

实际部署中存在的问题

决策延迟长　资源消耗高　管理能力差

- 网络系统的故障检测问题已经是几十年来网络研究者们一直尝试去解决的问题，现在挑战仍然很大。
- 深度神经网络经常包含数万个神经元，使得故障难以准确定位。

实际部署中存在的问题

决策延迟长　资源消耗高　管理能力差　可解释性低

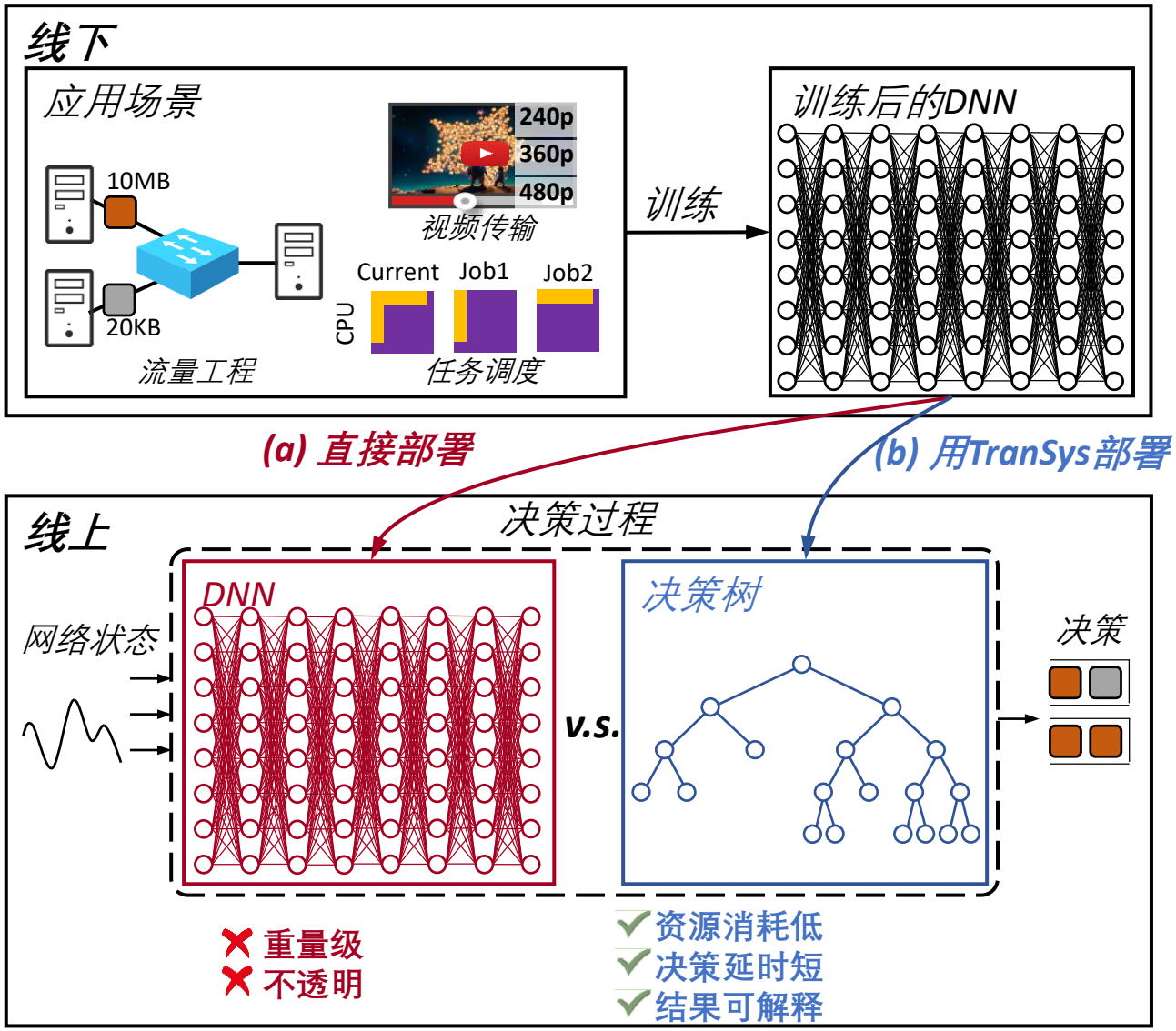
- 对于网络系统而言，一般存在一些性能比较好且简单的传统方法，如FIFO、SJF等。
- 即使引入神经网络提高了网络系统的性能，网络管理员也很难知道这一性能提升从何而来。

实际部署中存在的问题

决策延迟长　资源消耗高　管理能力差　可解释性低

重量级　不透明

智能网络系统在网络中实际部署障碍很多

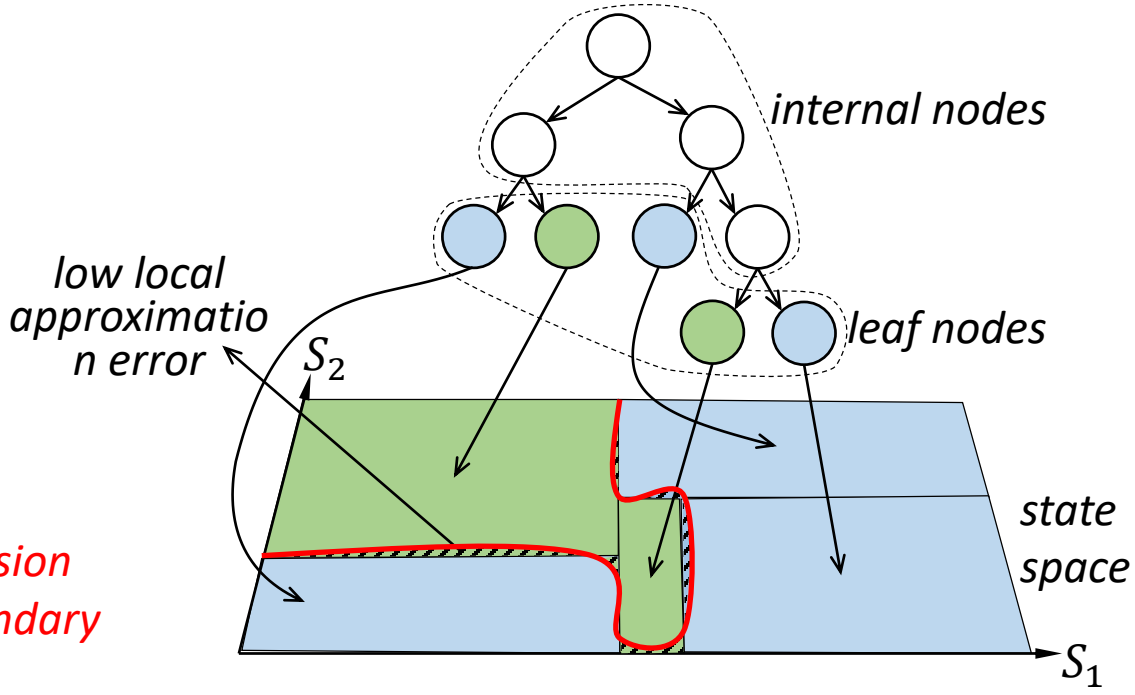


TranSys

将神经网络转换为决策树

为什么用决策树？

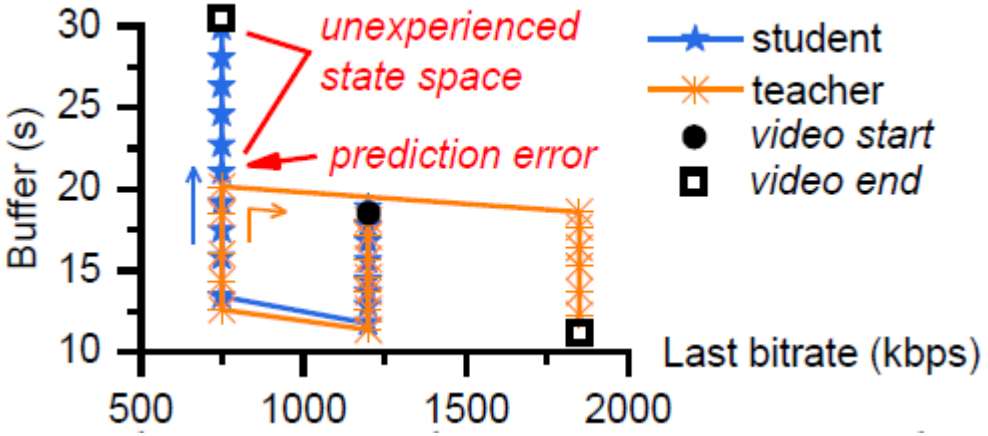
- 表达能力强
  - 尽管在序贯决策过程中决策树难以优化、训练，但其表达能力强 [NeurIPS'18].



为什么用决策树？

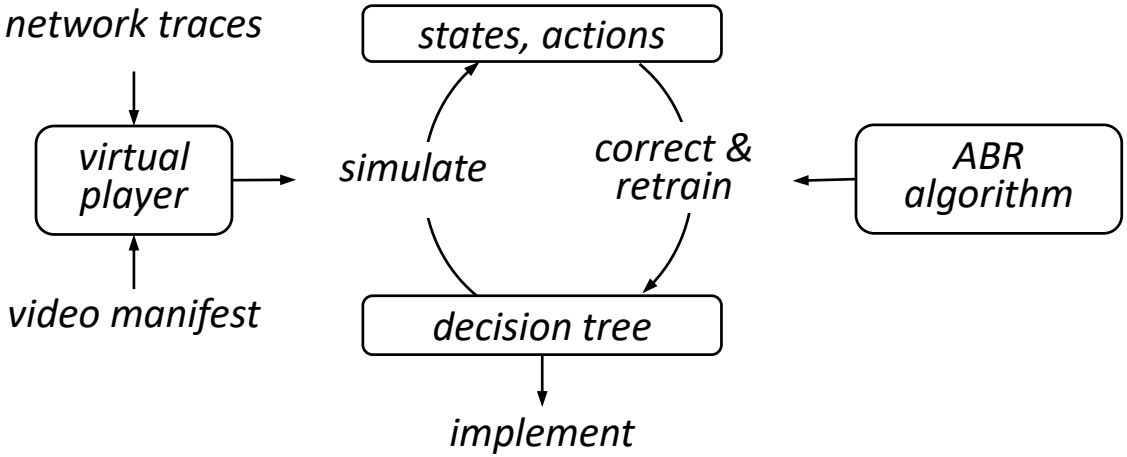
- 表达能力强
  - 尽管在序贯决策过程中决策树难以优化、训练，但其表达能力强 [NeurIPS'18].
- 足够轻量级
  - 二叉决策树由大量分支逻辑构成，无论是计算延时、还是物理可实现性、或者是资源消耗都很小。
- 与网络系统策略类似
  - 网络系统现有策略均是从不同方面共同做出判断。
  - 如视频传输，要求缓冲区不能过小、比特率变化不能过于剧烈、视频比特率尽可能高三者尽量同时满足。

- 网络系统：序贯决策过程的前后依赖性
  - 以视频传输为例，简单拟合决策树的话，一个错误决策可能会导致后续决策接连错误



student: 决策树
teacher: 神经网络

- 网络系统：序贯决策过程的前后依赖性
  - 以视频传输为例，简单拟合决策树的话，一个错误决策可能会导致后续决策接连错误
  - 解决方案：Follow-the-Leader类 模仿学习算法 [AIStats'11]



实验目标

- 保 真 度 ： TranSys能否保证在将神经网络转换为决策树的过程中，系统性能尽量不损失？
- 决策延迟： TranSys能有效降低多少决策延迟，带来什么效果？
- 资源消耗： TranSys能节约多少资源？有什么意义？
- 透 明 性 ： TranSys能否达到我们预期的透明性？我们用此可以做些什么？
- 部署代价： 网络管理员在部署TranSys的时候，需要付出多少额外部署代价？

部署细节——Pensieve与ToP

- ToP: TranSys over Pensieve
- Apache视频服务器，Chrome客户端浏览器。
- 六种比特率：{300, 750, 1200, 1850, 2850, 4300}kbps
- 两个数据集：挪威HSDPA 3G流量，美国FCC宽带流量
- 五个基线方案：BB[SIGCOMM'14], RB[MMSys'11], FESTIVE[CoNEXT'12], BOLA[INFOCOM'16], rMPC[SIGCOMM'15]
- 决策树直接用JavaScript写出，原Pensieve的神经网络在客户端上采用Tensorflow.js[SysML'19]实现

部署细节——AuTO与ToA

ToA: TranSys over AuTO

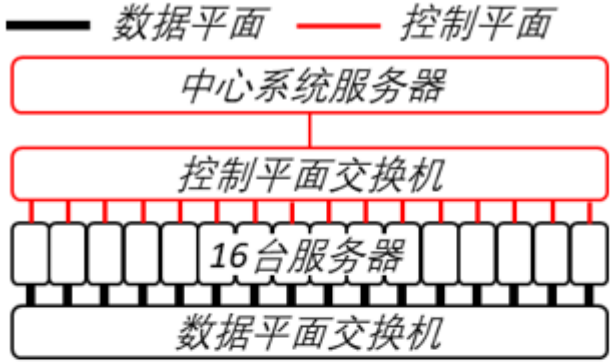


图 4.2 AuTO 部署拓扑结构

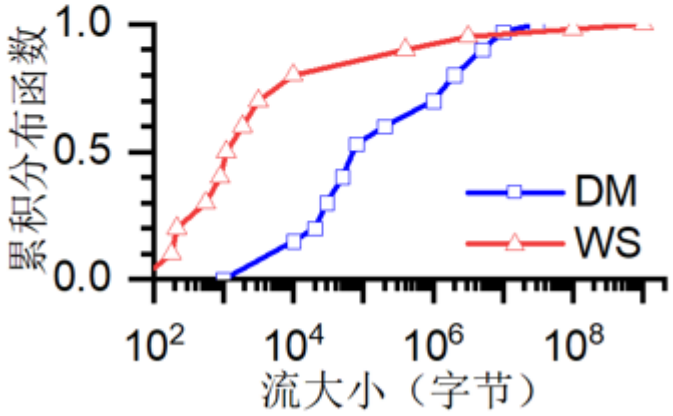
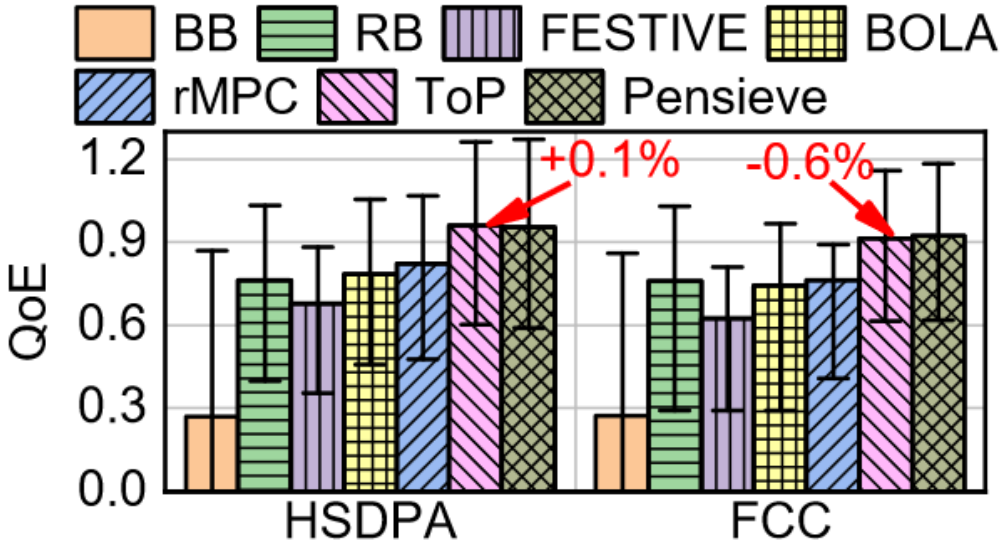
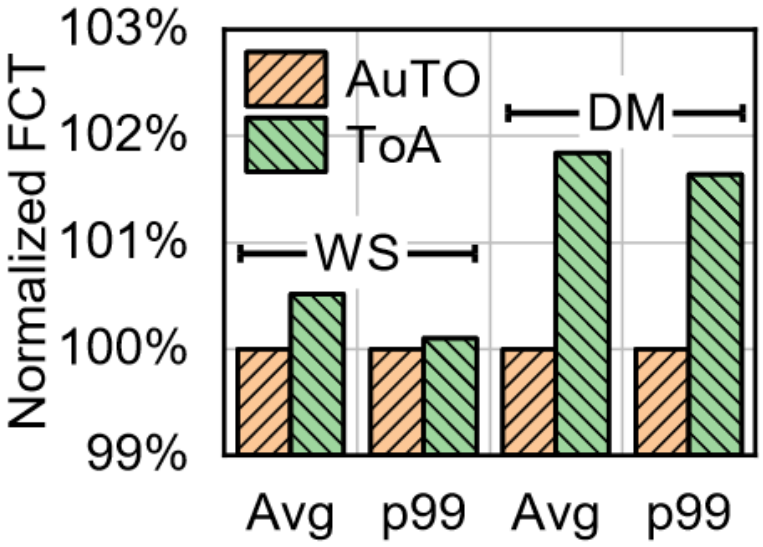


图 4.3 AuTO 部署中采用的数据流量特征

保真度



(a) TranSys over Pensieve. Error bars represent 25%ile and 75%ile.



(b) TranSys over AuTO.

决策延迟

- JavaScript中部署的实际决策延时（ToP）
  - PC: Intel Core i7-8700 CPU
  - Mobile: 高通骁龙710

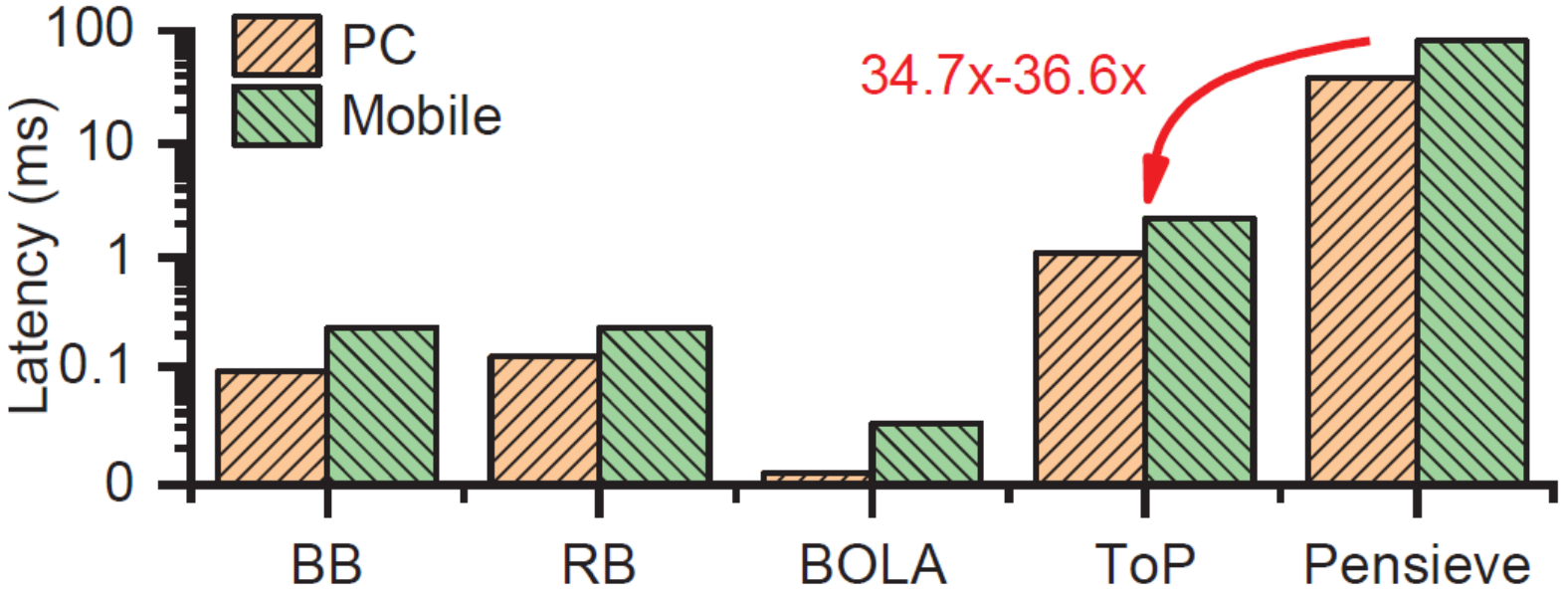


Figure 10: Decision latency on different end devices.

决策延迟

对ToA系统，决策延迟短可以带来：

- 性能收益：中间流8%完成时间优化。
- 部署收益：可部署到网卡上。我们基于智能网卡P4语言实现的决策树延迟为9.37μs。

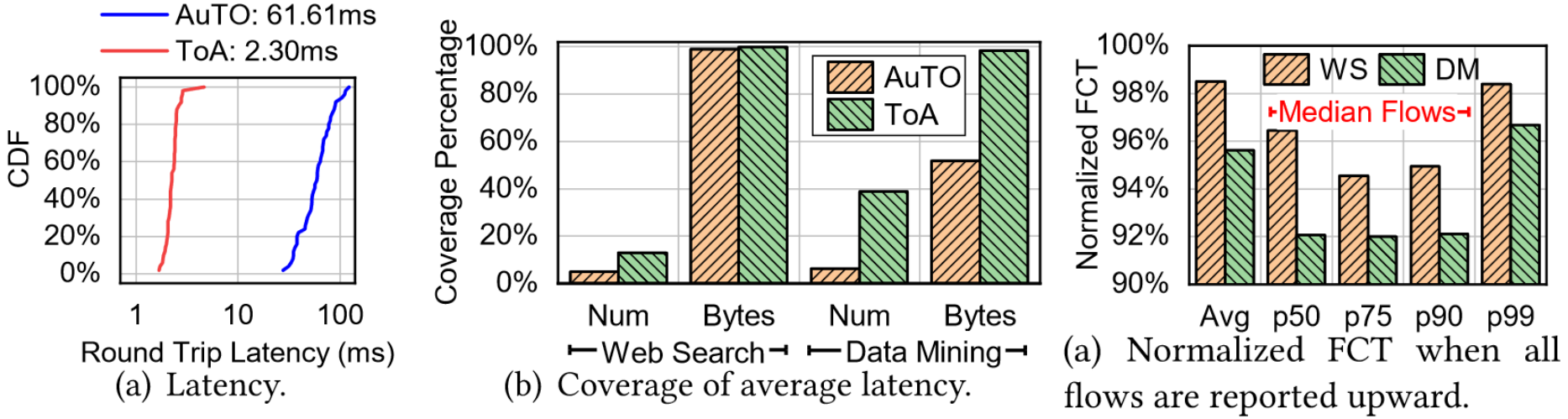
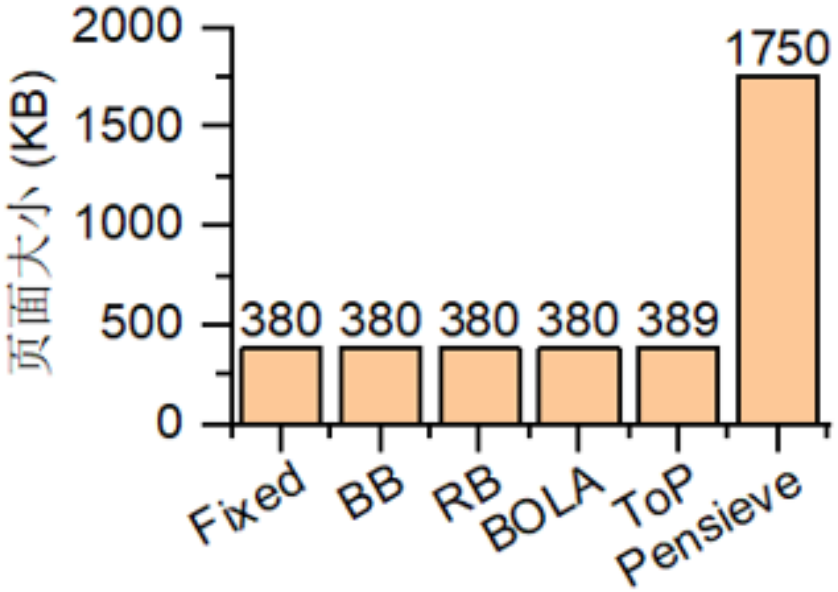
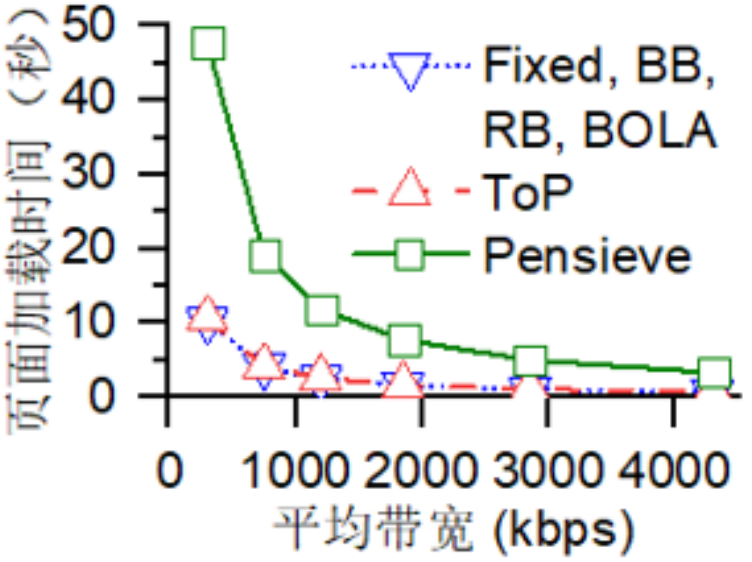


Figure 4: The latency between flow servers send states to and receive actions from the RL server.

资源消耗



(a) 页面大小



(b) 页面加载时间

$$\frac{\Delta Pensieve}{\Delta ToP} = \frac{1750KB - 380KB}{389KB - 380KB} = 156$$

资源消耗

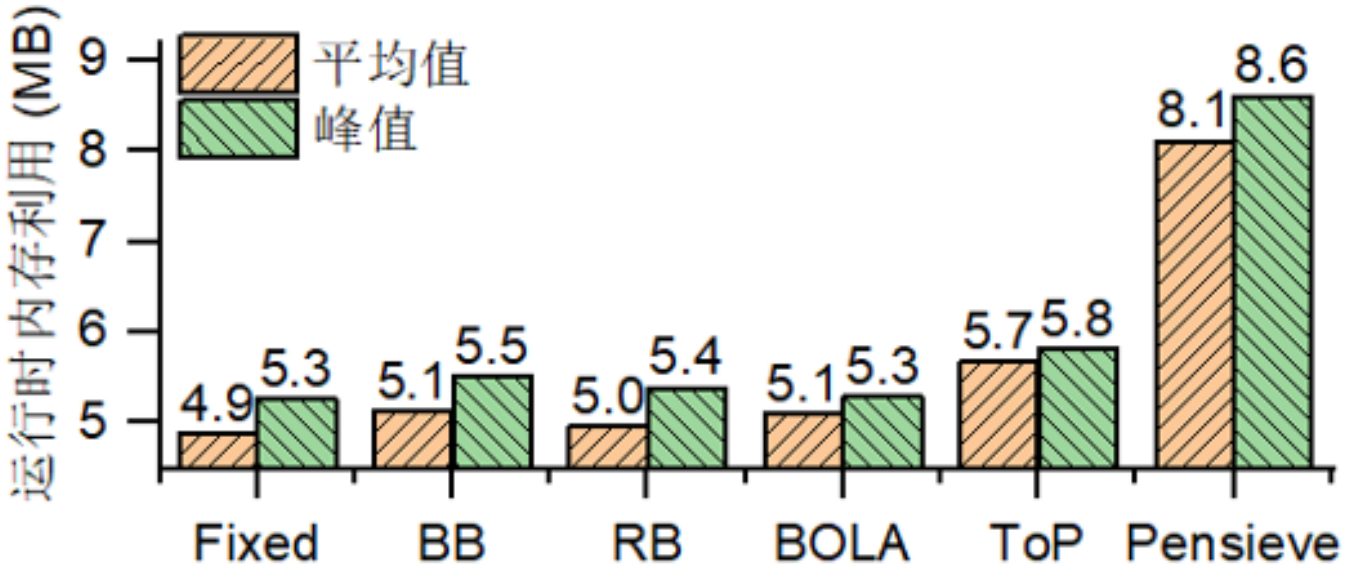


图 4.11 运行时 JavaScript 内存消耗。固定比特率算法的内存消耗是由 JavaScript 中的其他功能造成的。

透明性——故障检测

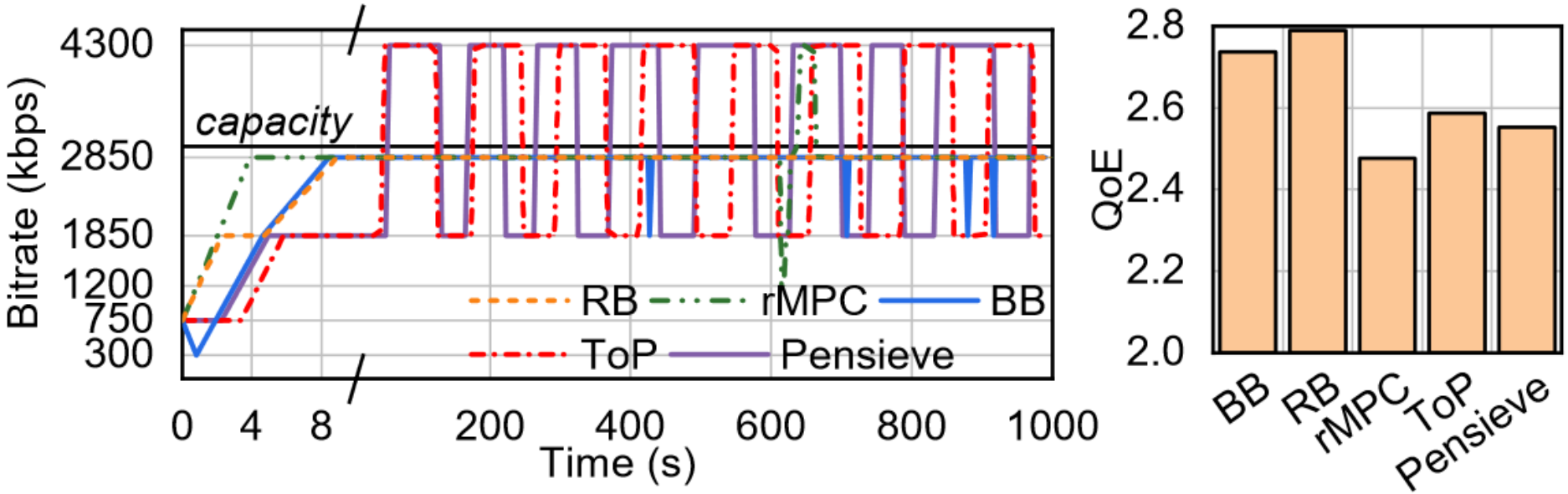


Figure 6: Bitrate decisions on a 3000kbps link. Six bitrate levels from 300kbps to 4300kbps are marked on the y-axis of the left figure. BB, RB and rMPC converge to 2850kbps while Pensieve and ToP oscillate between 1850kbps and 4300kbps.

透明性——故障检测

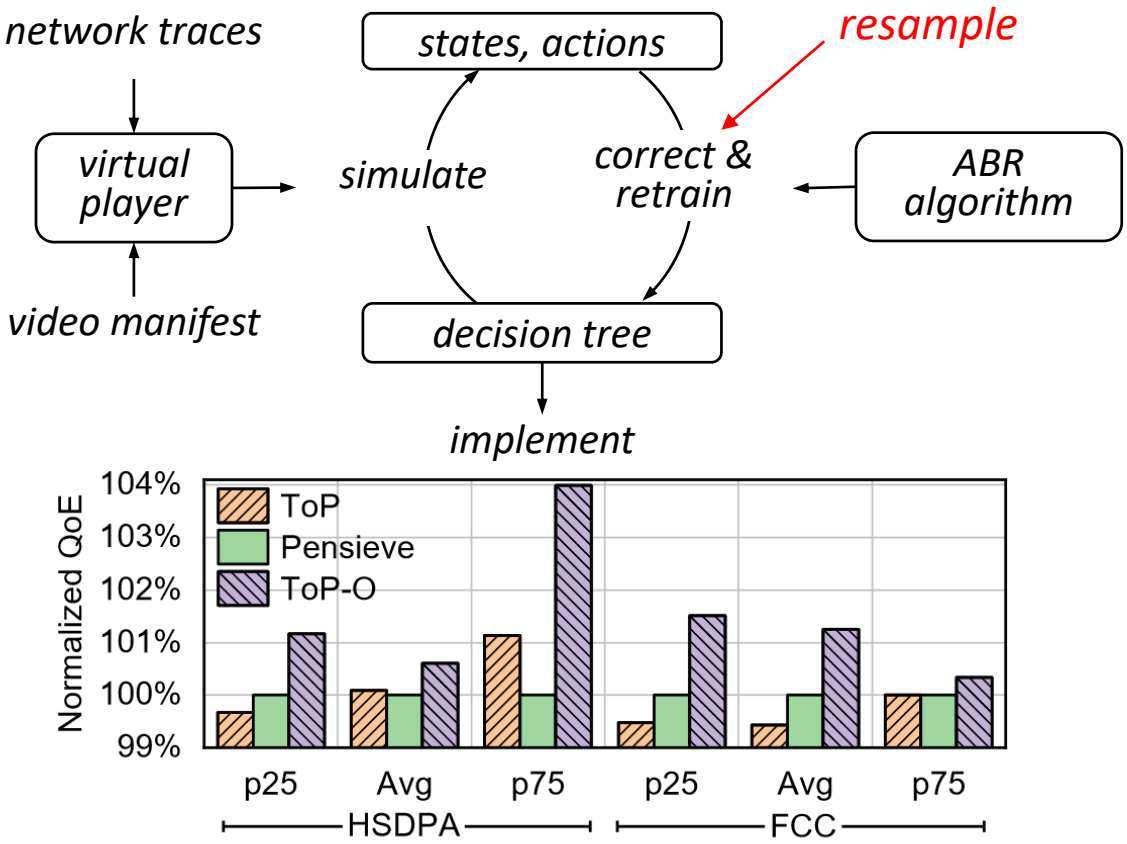


Figure 7: QoE of ToP with oversampling (ToP-O) normalized by Pensieve.

透明性——策略解释

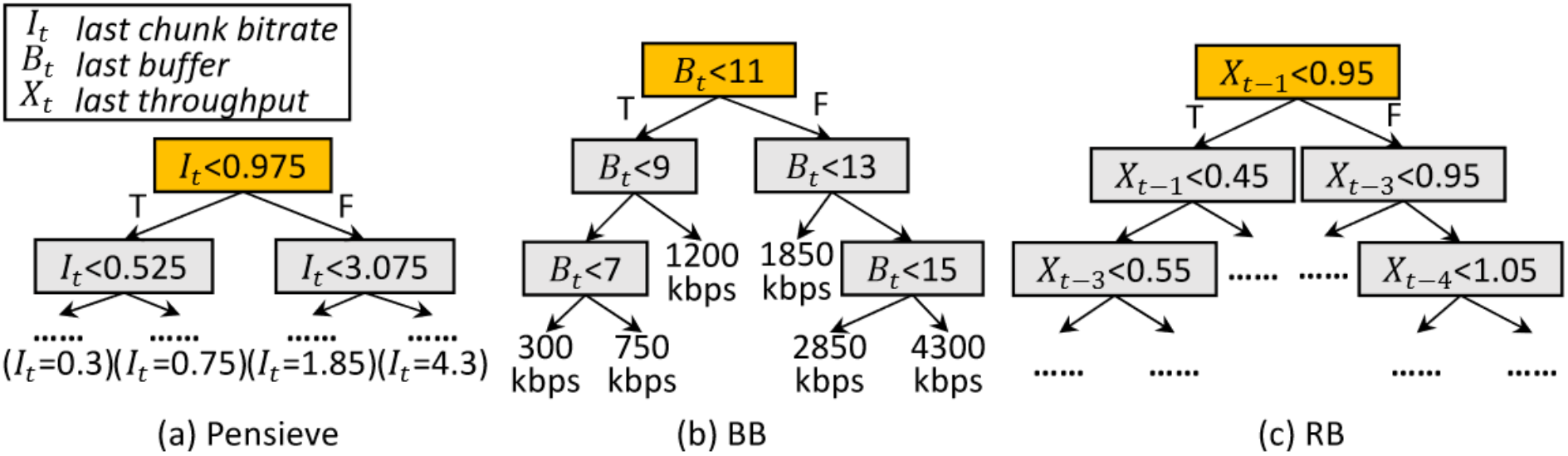
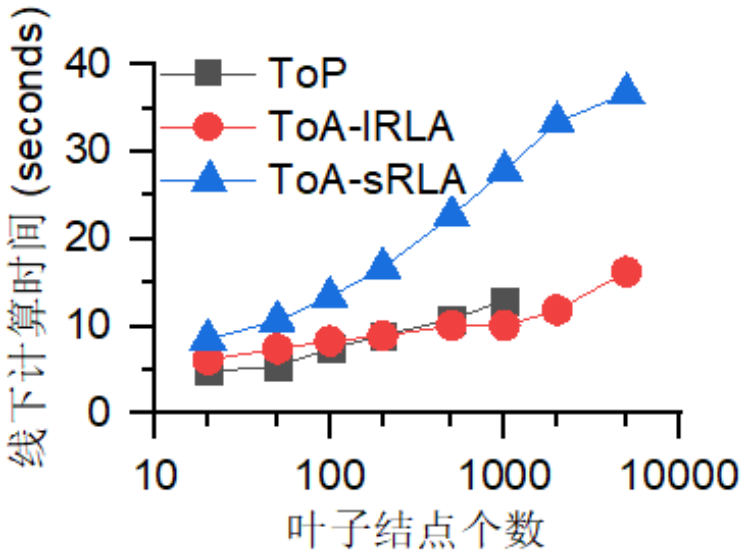


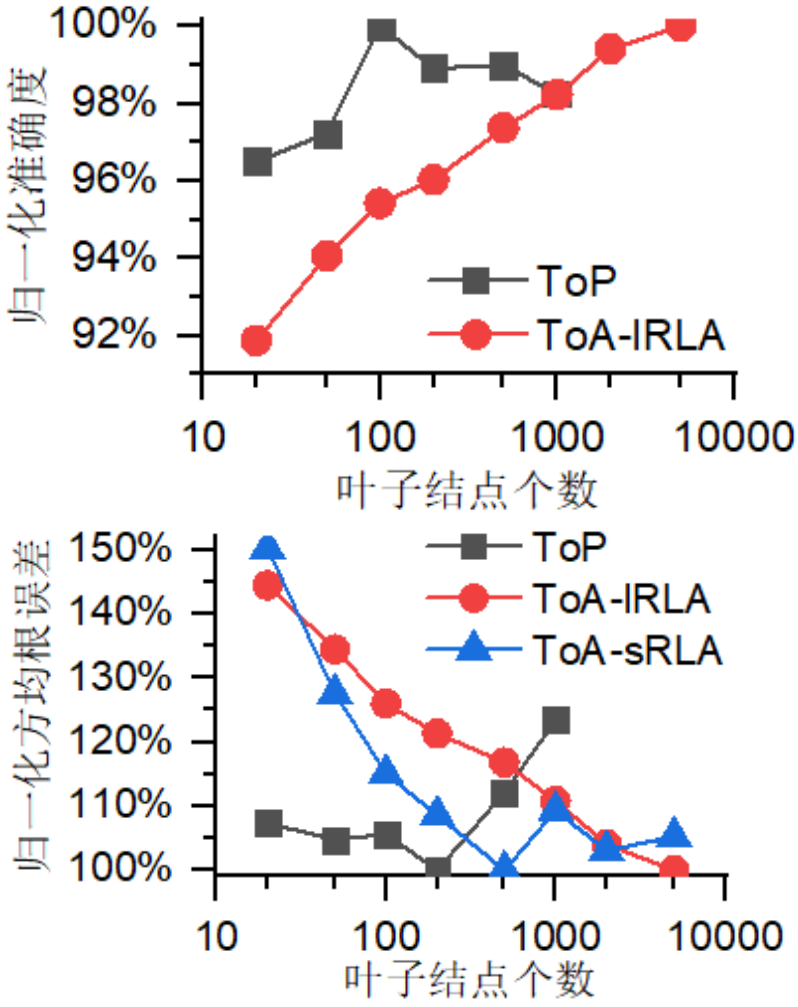
Figure 8: Decision tree representations of ABR algorithms.

部署代价

线下计算时间



敏感性分析



学期	月份	工作内容
秋季学期	11	学习网络技术相关基础知识、调研智能网络系统
	12	总结并初步验证其存在的实用性和可解释性问题
	1	进行初步实验，通过线性拟合等方式尝试解决，并通过仿真环境进行初步实验比选
	2	结合初步实验结果，基本确定采用决策树方案
春季学期	3	搭建实验网络拓扑、配置实验环境、搭建模型转换系统
	4	根据真实实验中发现的问题迭代设计，完善工作
	5	进行补充实验，进一步验证设计
	6	整理实验资料，完成综合论文的撰写与答辩工作

结语

- 总结了目前智能网络系统存在的重量级、不透明两大实用性问题。
- 提出了在网络系统部署时用决策树来代替神经网络的思路。
- 通过两个智能网络系统的实验，取得如下效果：
  - **高保真**：性能损失<1.8%。
  - **低延迟**：决策延迟降低1-2个数量级，带来收益。
  - **低消耗**：资源大幅降低（额外页面大小降低156x）。
  - **透明性**：可用于故障检测与策略解释。
  - **代价小**：管理员付出额外代价小。

未来工作

- 正在整理项目开源，将发布在https://transys.io
- 将TranSys系统部署于更多的智能网络系统上：
  - 集群调度（Decima[SIGCOMM'19]）。
  - 拥塞控制（TCP-RL[JSAC'19]）。
- 智能网络系统的其他实用性问题：
  - 公平性。
  - 动态性。
  - 安全性。
  - 可扩展性。

清華大學
Tsinghua University

谢谢聆听，欢迎提问！

综合论文训练答辩

无58 孟子立

补充

相关工作

- 将DNN轻量化
  - 利用远程专有服务器加速，如MCDNN[MobiSys'16]，但延迟高、成本高。
  - 利用专有硬件加速，如用GPU、DSP、ASIC等等，但成本高、可扩展性差。
  - 神经网络剪枝压缩，如NestDNN[MobiCom'18]，但效果有限、性能下降。
- 可解释性研究
  - 网络安全：AI^2 [S&P'18], LEMNA[CCS'18]
  - 图像分析：NetDissect[CVPR'17], ICNN[CVPR'18]
  - 推荐系统：Narre[WWW'18]

远程服务器成本高

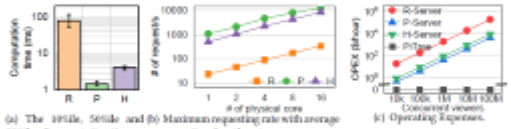


Figure 1: Load testing results of remote ABR servers. [R, P, H] refer to [RobustMPC [65], Pensieve [47], HotDASH [53]]. We build ABR servers with tornado [6] and test the capacity with vegeta [7] from another directly-connected server.

ToP输入流量与实验指标

Table 2: QoE metrics considered in our evaluation [47, 64]

QoE_{lin}	$q(R) = R$	μ=4.3
QoE_{log}	$q(R) = \log(R/R_{min})$	μ=2.66
QoE_{hd}	$q(R) = \{0.3, 1, 0.75, 2, 1.2, 3, 1.85, 12, 2.85, 15, 4.3, 20\}$	μ=8

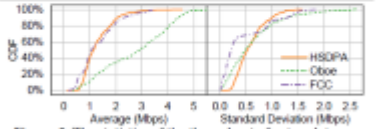


Figure 5: The statistics of the throughput of network traces.

ToP泛化能力

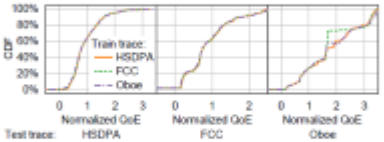
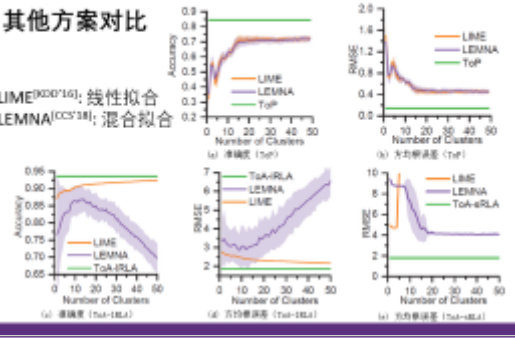


Figure 11: Normalized QoE_{lin} of PiTree-over-Pensieve when test traces are statistically different from training traces.

其他方案对比

LIME[KDD'16]: 线性拟合
LEMNA[CCS'18]: 混合拟合



其他两种ABR算法的补充实验

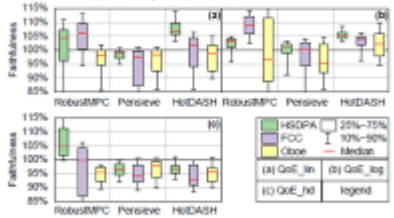
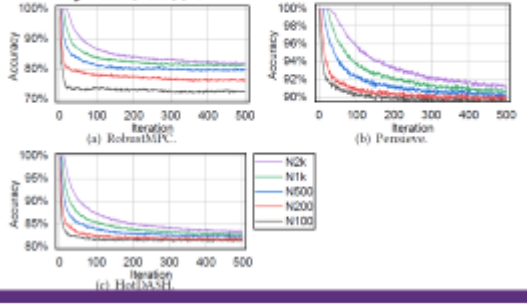
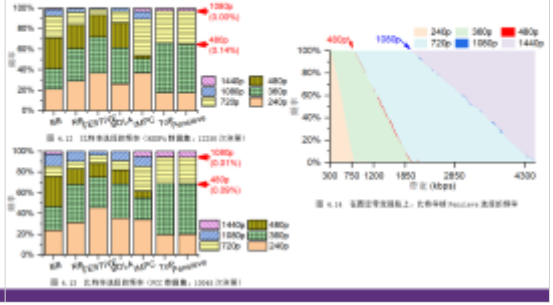


Figure 7: Faithfulness of PiTree on different ABR algorithms, network traces and QoE metrics.

TranSys训练鲁棒性



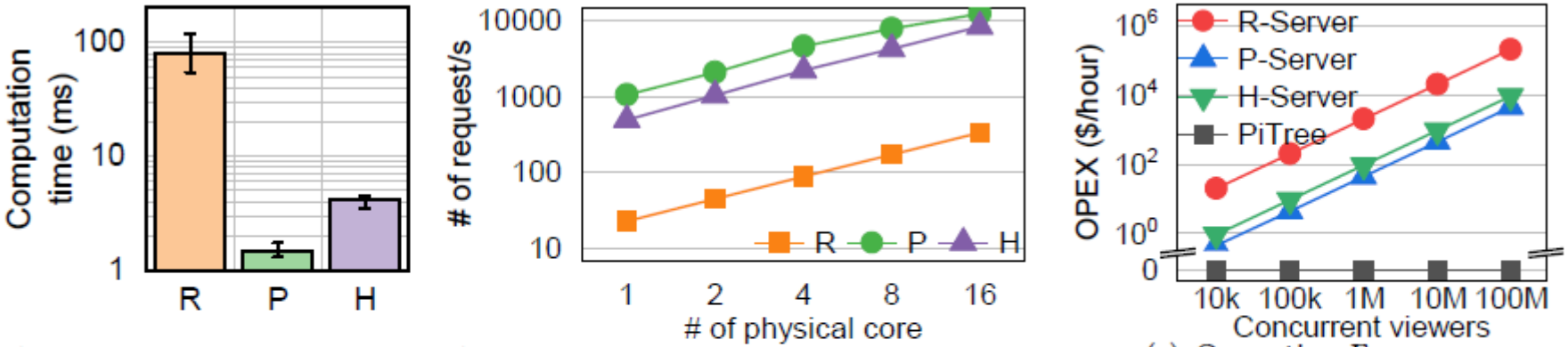
不均衡策略选择



相关工作

- 将DNN轻量化
  - 利用远程专有服务器加速，如MCDNN[MobiSys'16]，但延迟高、成本高。
  - 利用专有硬件加速，如用GPU、DSP、ASIC等等，但成本高、可扩展性差。
  - 神经网络剪枝压缩，如NestDNN[MobiCom'18]，但效果有限、性能下降。
- 可解释性研究
  - 网络安全：AI^2 [S&P'18], LEMNA[CCS'18]
  - 图像分析：NetDissect[CVPR'17], ICNN[CVPR'18]
  - 推荐系统：Narre[WWW'18]

远程服务器成本高



(a) The 10%ile, 50%ile and 90%ile of computation time.

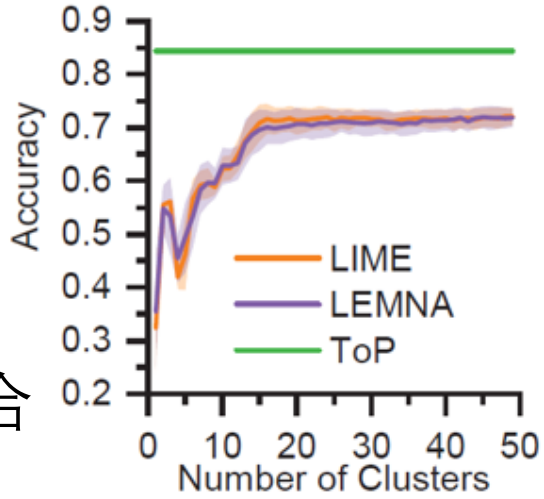
(b) Maximum requesting rate with average response time less than 1s.

(c) Operating Expenses.

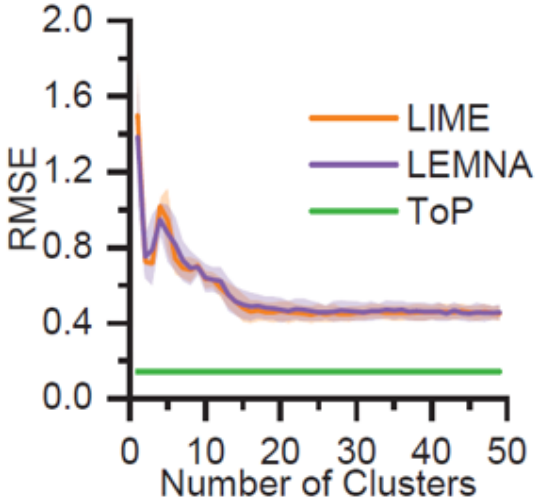
Figure 1: Load testing results of remote ABR servers. {R, P, H} refer to {RobustMPC [65], Pensieve [47], HotDASH [53]}. We build ABR servers with `tornado` [6] and test the capacity with `vegeta` [7] from another directly-connected server.

其他方案对比

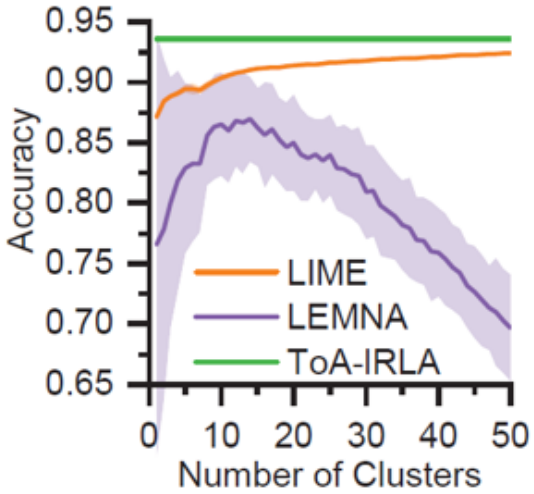
LIME[KDD'16]: 线性拟合
LEMNA[CCS'18]: 混合拟合



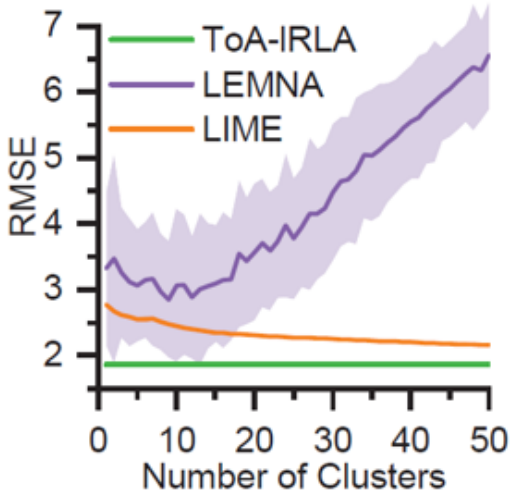
(a) 准确度（ToP）



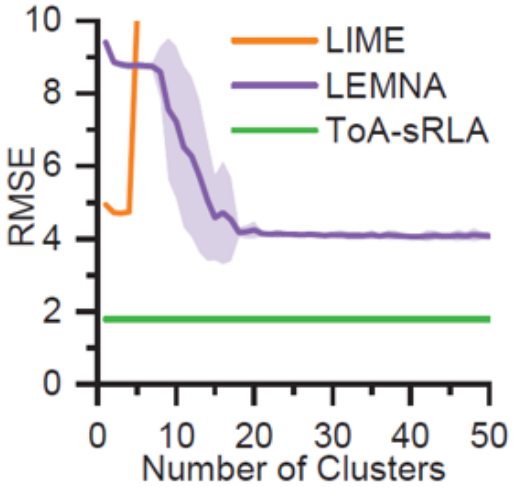
(b) 方均根误差（ToP）



(c) 准确度（ToA-IRLA）



(d) 方均根误差（ToA-IRLA）



(e) 方均根误差（ToA-sRLA）

不均衡策略选择

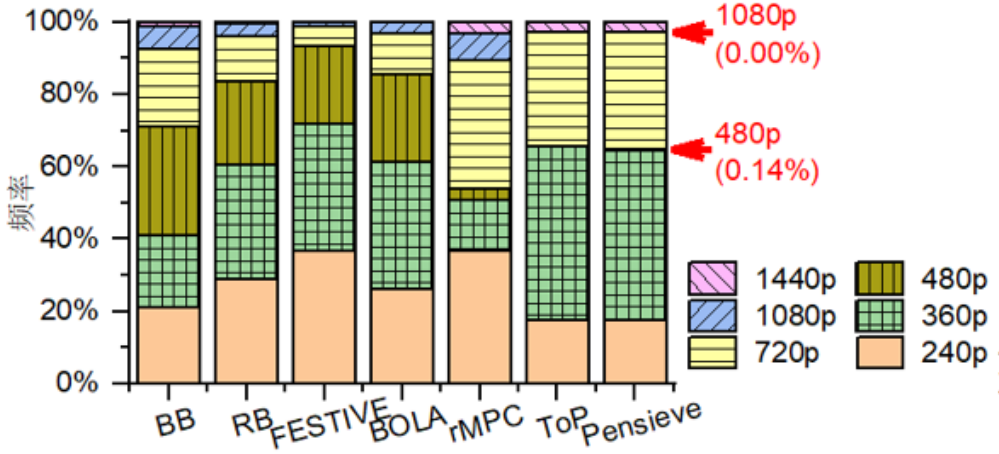


图 4.12 比特率选择的频率（HSDPA 数据集，12250 次决策）

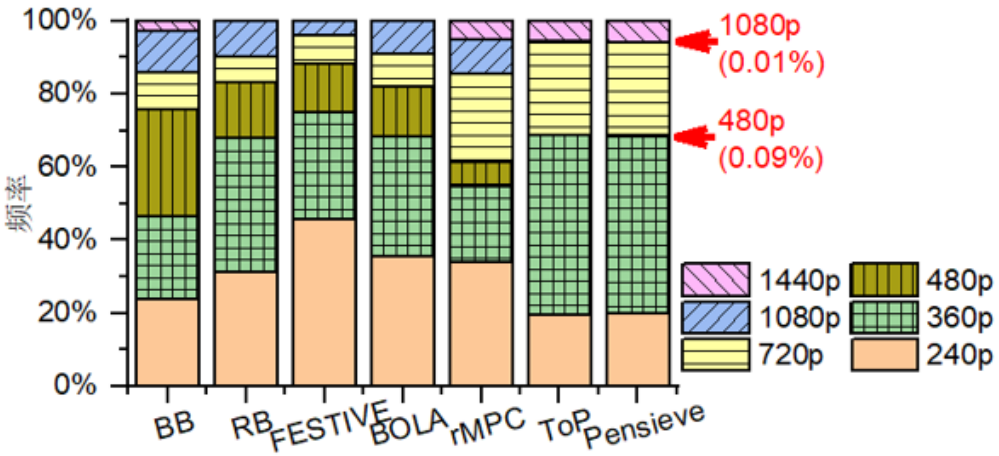


图 4.13 比特率选择的频率（FCC 数据集，10045 次决策）

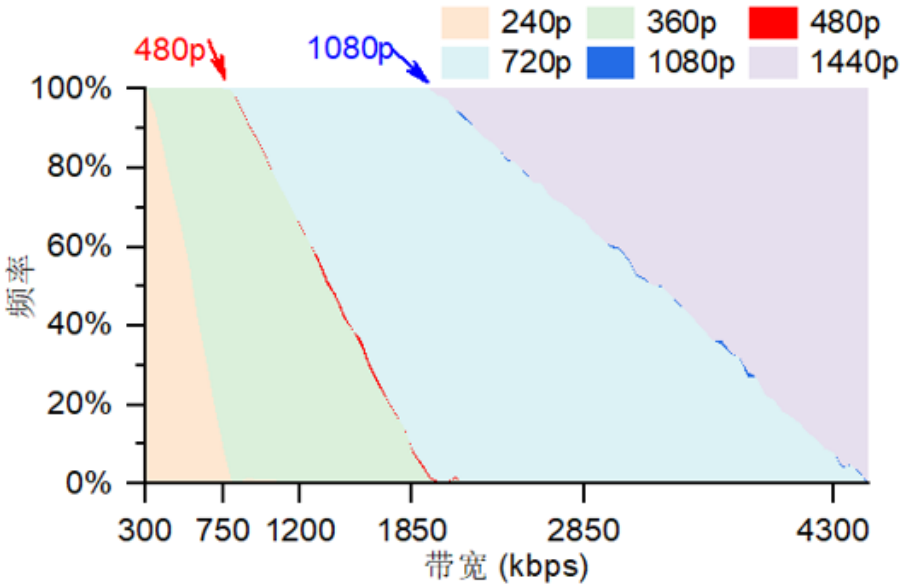


图 4.14 在固定带宽链路上，比特率被 Pensieve 选择的频率

ToP输入流量与实验指标

Table 2: QoE metrics considered in our evaluation [47, 64]

QoE_{lin}	$q(R) = R$	μ=4.3
QoE_{log}	$q(R) = \log(R/R_{min})$	μ=2.66
QoE_{hd}	$q(R) = \{0.3{:}1, 0.75{:}2, 1.2{:}3, 1.85{:}12, 2.85{:}15, 4.3{:}20\}$	μ=8

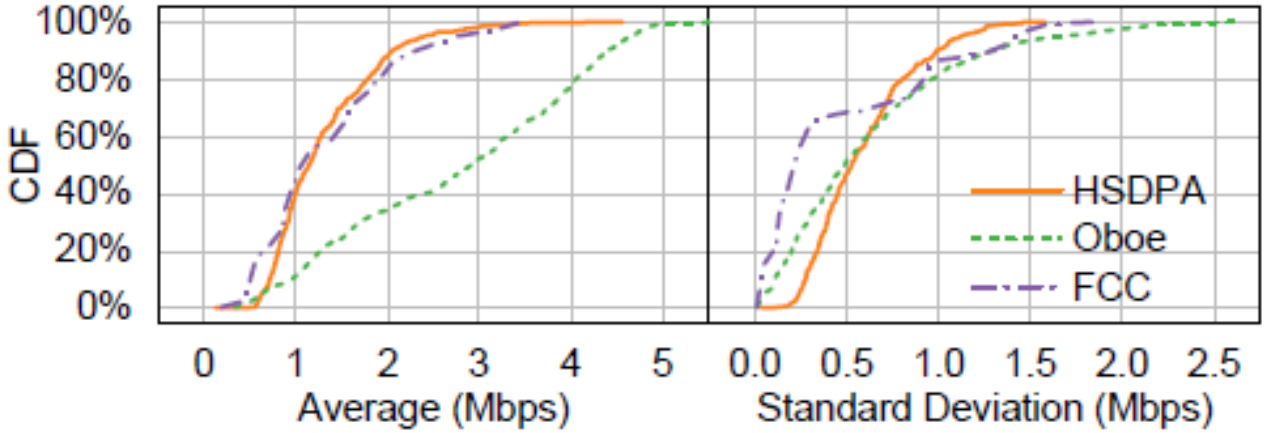


Figure 5: The statistics of the throughput of network traces.

其他两种ABR算法的补充实验

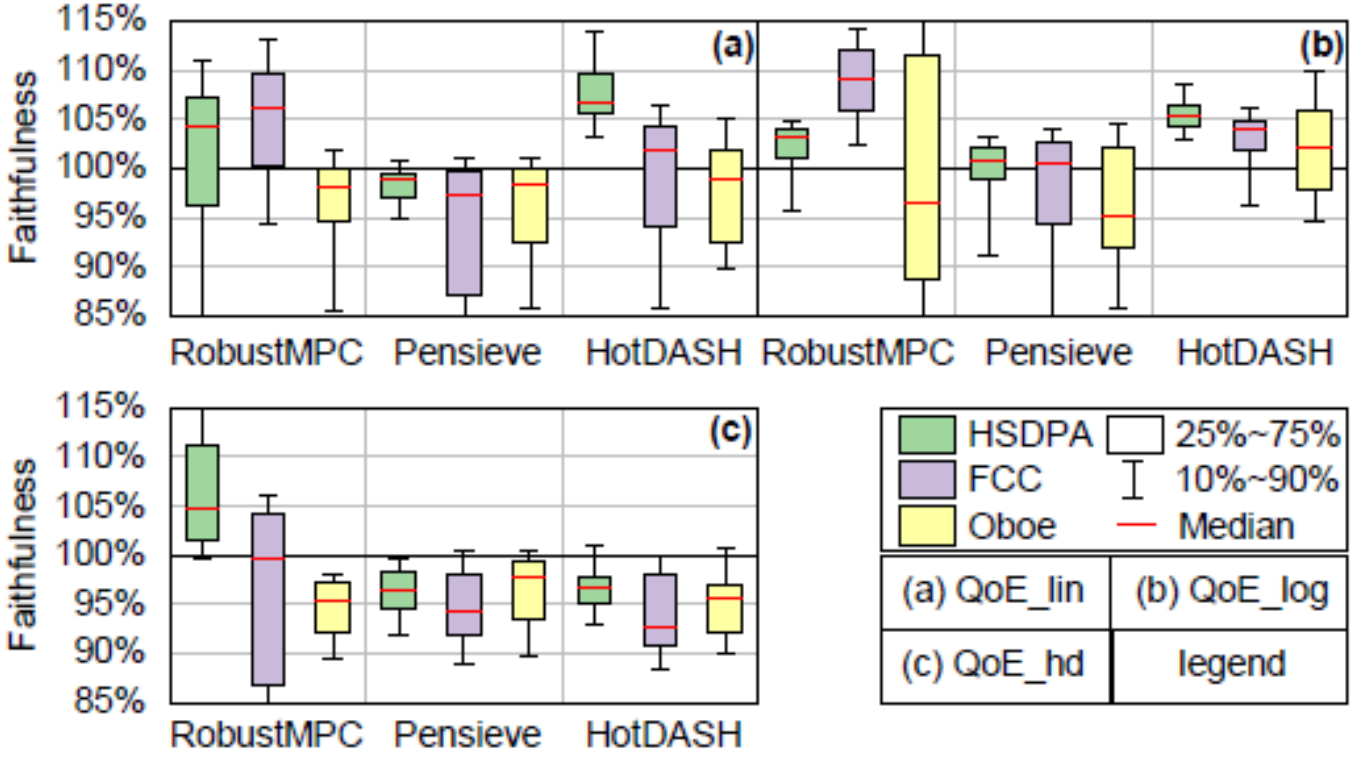


Figure 7: Faithfulness of PiTree on different ABR algorithms, network traces and QoE metrics.

ToP泛化能力

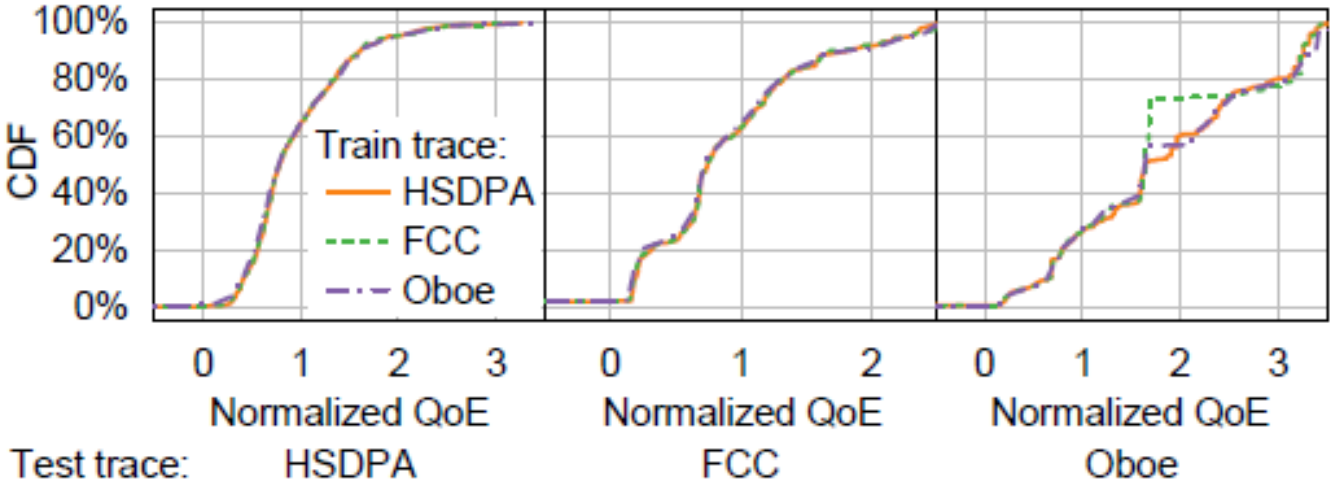
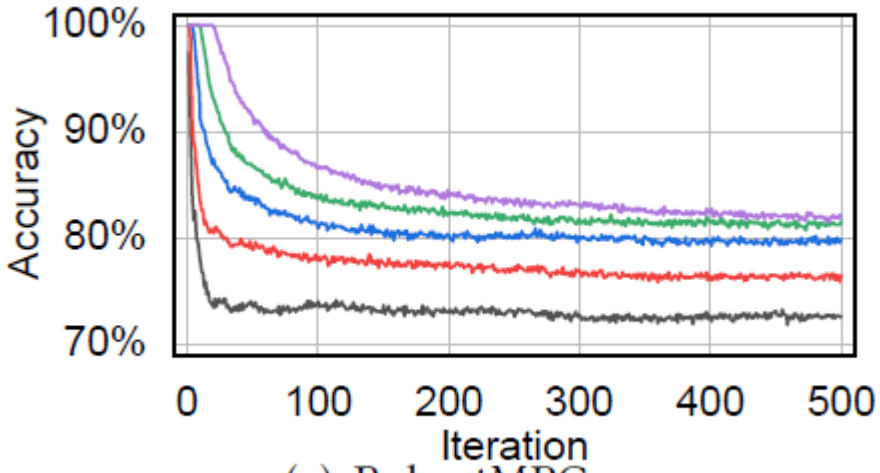
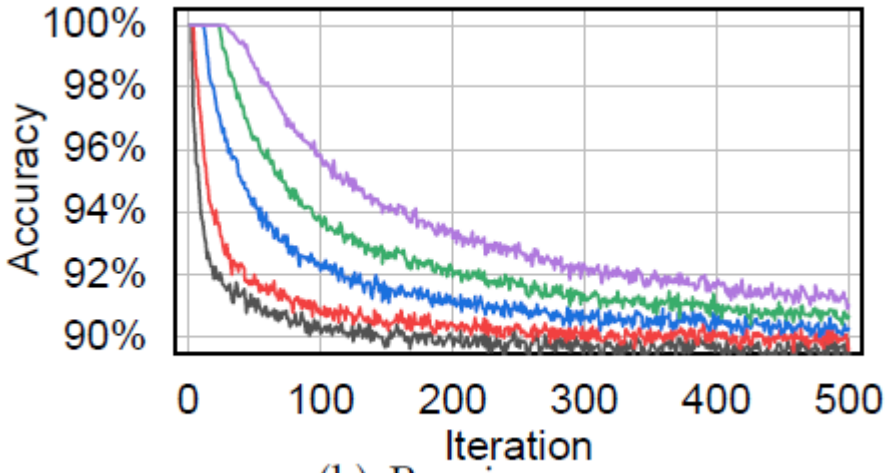


Figure 11: Normalized QoE_{lin} of PiTree-over-Pensieve when test traces are statistically different from training traces.

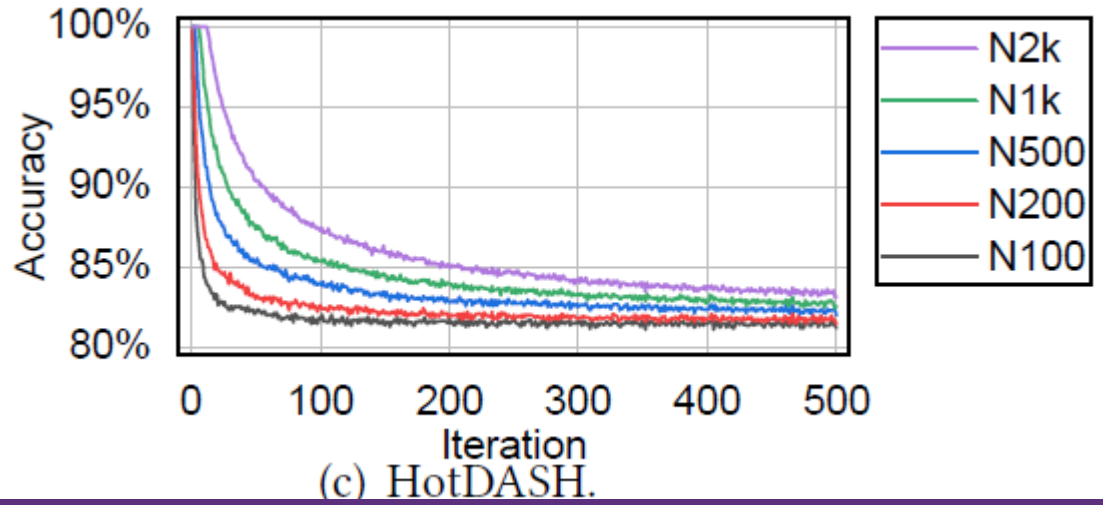
TranSys训练鲁棒性



(a) RobustMPC.



(b) Pensieve.



(c) HotDASH.