

To Adapt or Not to Adapt, That is the Ski Question

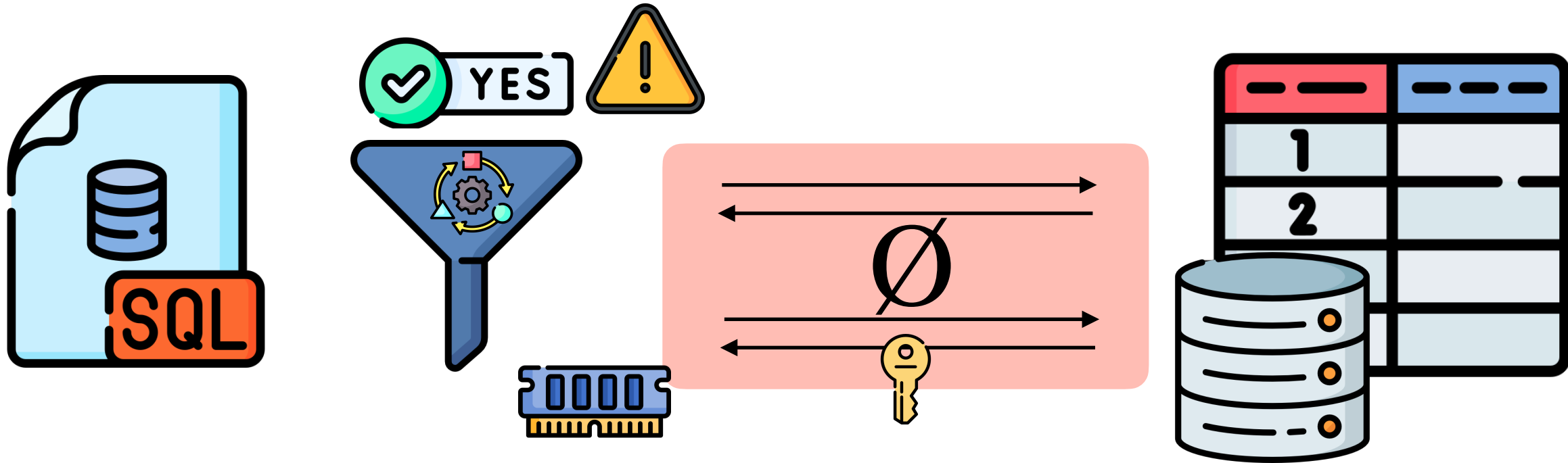
Building online adaptive filters for robust performance

Yuvaraj Chesetti, Prashant Pandey



Background

Adaptive filters learn from their false positive mistakes



Is X in set? YES (**False positive**)
Is X in set? NO (**True negative**)
Is X in set? NO (**True negative**)

Adaptive filters bound the number of false positives to close to ϵn

Problem

Adaptive filters increase the I/O overhead of false positives

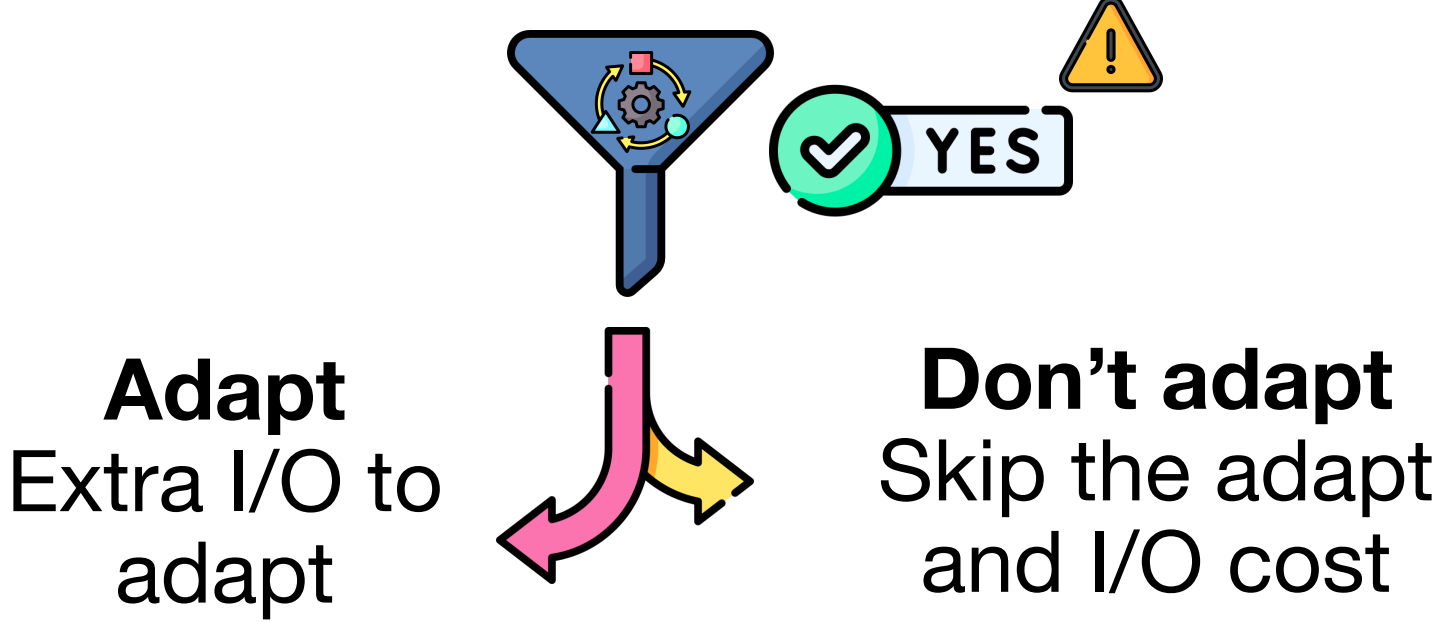
	Non-Adaptive	Adaptive
Disk I/O cost	1	$(1 + \alpha)$
Uniform	ϵn	$(1 + \alpha) \cdot (\epsilon n)$
Skewed	$\leq n$	$(1 + \alpha) \cdot (\epsilon n)$

α : relative extra I/O to fix false positive

ϵ : false positive probability of single query

n : number of queries

Online Adaptivity



HybridSkiQF



Analyze false positive repetitions in window

Window decision

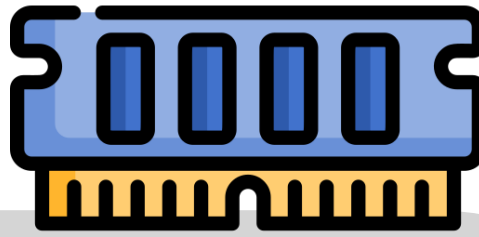
Never adapt

Always adapt

Revisit decision after window ends



The Ski-Adaptive Quotient Filter (SkiQF)



Adaptivity as a ski-rental problem

Buying $\rightarrow$ Adapting
Renting $\rightarrow$ Not adapting

Buy when cumulative renting costs exceed buying cost

α : Overhead of adapting
 τ : When to adapt

α	τ	Comp. Ratio
≥ 1	$\lfloor \alpha \rfloor$	≤ 2
$[0.618, 1)$	1	$\frac{(2 + \alpha)}{(1 + \alpha)}$
$[0, 0.618)$	0	$(1 + \alpha)$

Set: {B,L,R}

$h(B)=001\ 100..$

$h(L)=100\ 101..$

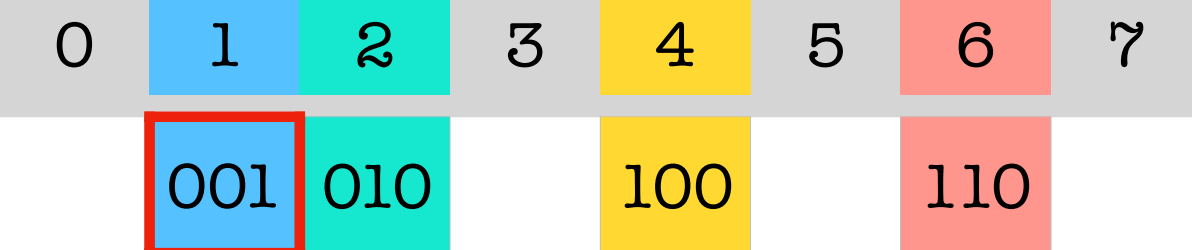
$h(R)=010\ 110..$

$h(F)=001\ 011..$



$h(B)=001\ 100..$

Design



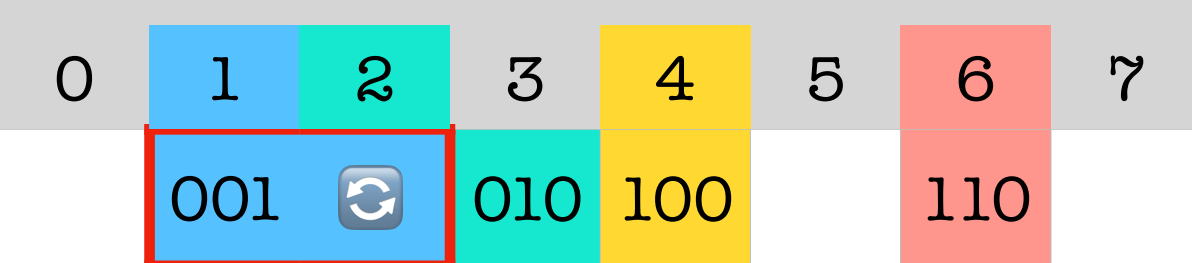
Linear probing hash-table of fingerprints. (Quotient Filter)

Is B in set $\rightarrow$ Is prefix of $h(B)$ in filter?

(True Positive)

Is F in set $\rightarrow$ Is prefix of $h(F)$ in filter?

(False Positive)

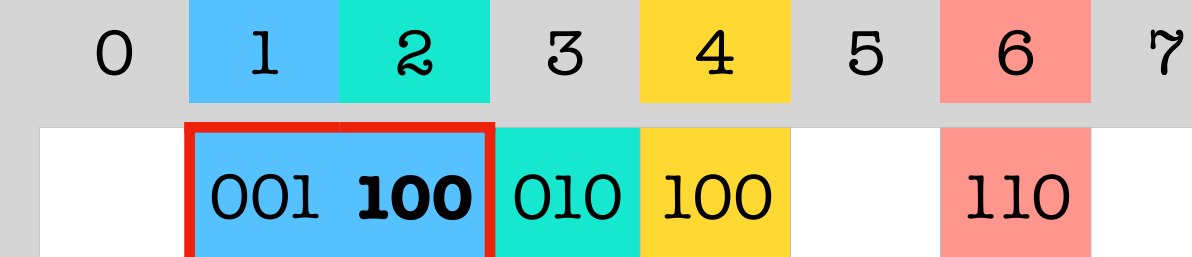


Counter slot initialized if empty

Has counter reached τ ?

Renting

Yes! Adapting...



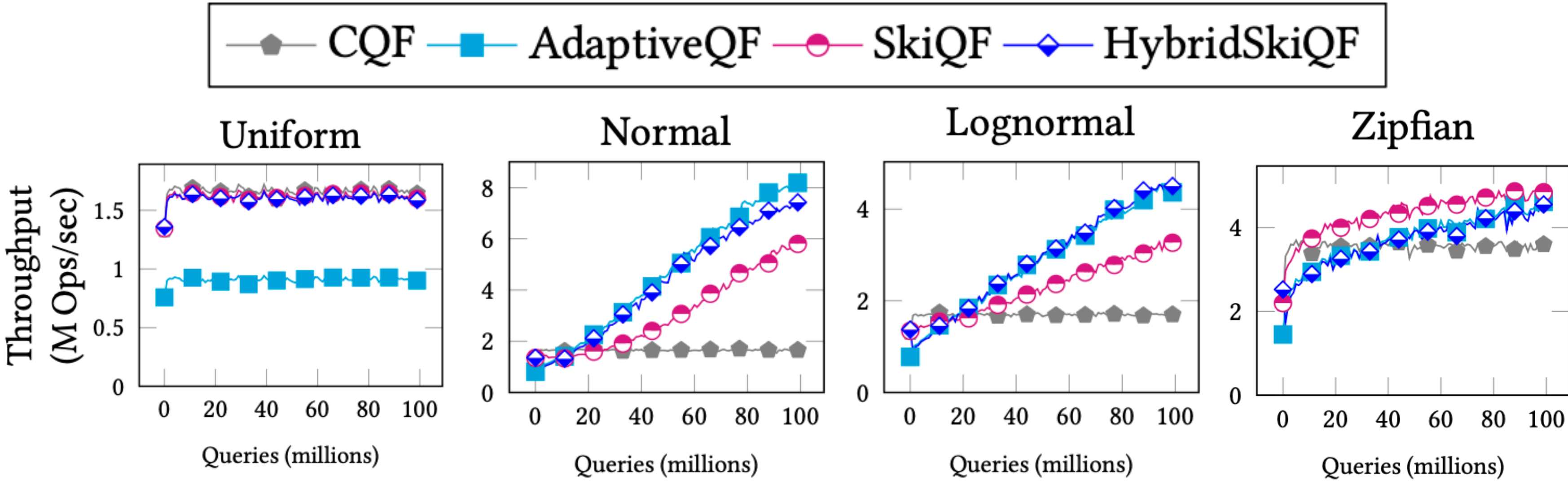
Adapt by extending fingerprint

Buying

Is F in set $\rightarrow$ Is prefix of $h(F)$ in filter?

(True Negative)

Experimental Results



Paper



Code



SkiQF reduces I/O overhead of adaptivity while maintaining competitive throughput across all workload distributions