Adaptive String Dictionary Compression in In-Memory Column-Store Database Systems

Ingo Müller, Cornelius Ratsch, Franz Faerber / KIT / SAP AG

March 26, 2014 – EDBT, Athens, Greece

Motivation: Column-Store Architecture Recap

Logical representation

First name
Helen
Michael
Michael
Adam
Michael
Helen
Adam

$\Longrightarrow$

Physical representation

Code
2
3
3
1
3
2
1

Column vector

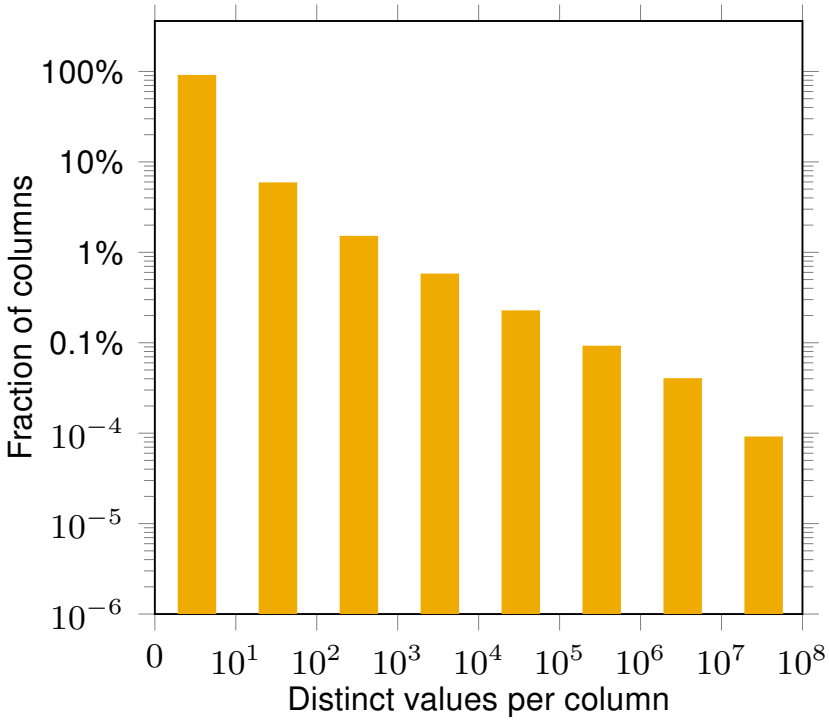
Code	Value
1	Adam
2	Helen
3	Michael

Dictionary

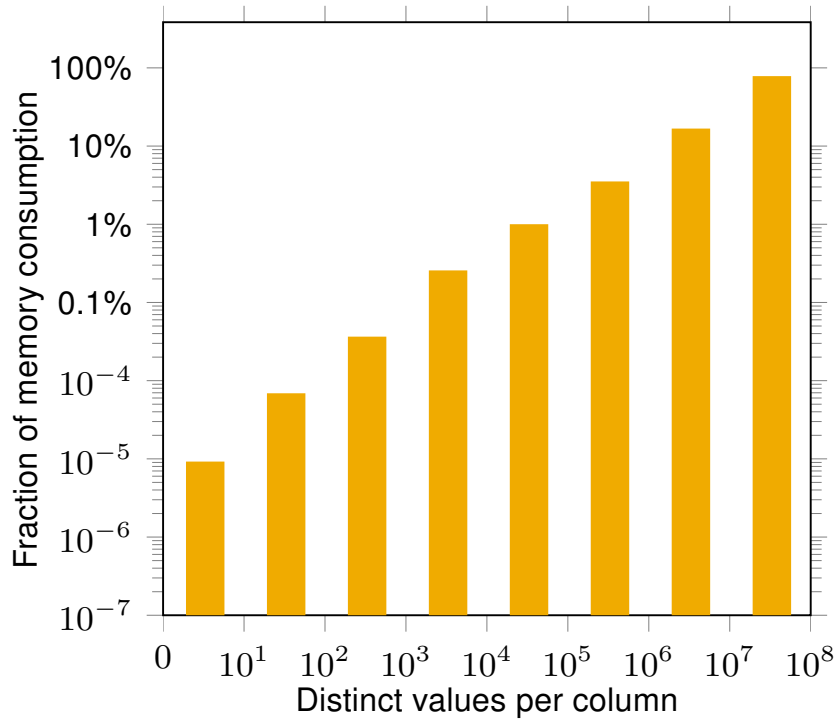
- Focus on static dictionaries of the read-optimized store

Motivation: Observation of Real-World Data

Distribution of cardinalities



Memory consumption



- String columns play an important role in real-world applications
  $\rightarrow$ potential for compression

Outline

1. Introduction
2. Survey of Dictionary Formats
3. Performance Models
4. Automatic Selection
5. Evaluation
6. Summary

Survey of Dictionary Formats

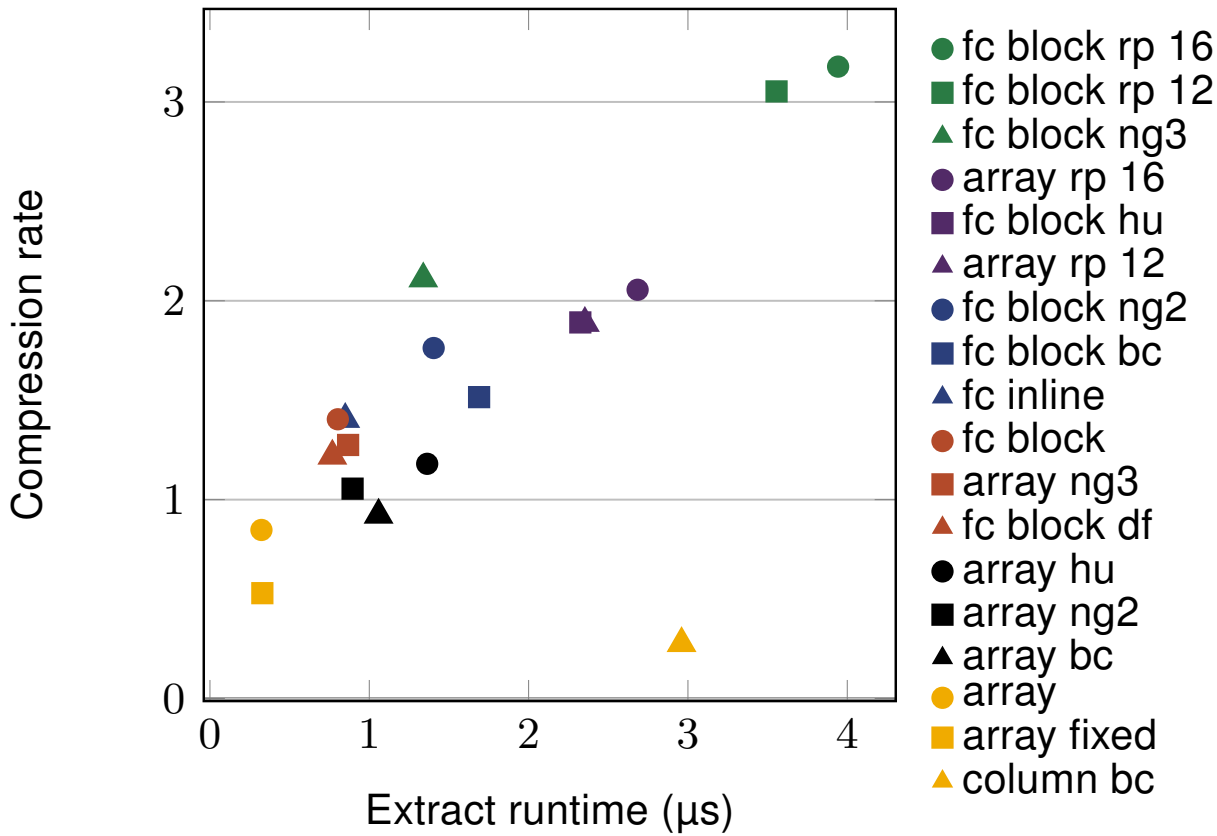
Two basic dictionary formats

- Array
- Front coding

Combined with string compression schemes

- Uncompressed
- N-gram compression
- Bit compression
- Huffman
- Re-Pair

Performance Comparison of Dictionary Formats



- Formats provide trade-off between runtime and space consumption
- Trade-off depends on dictionary content

Compression Models

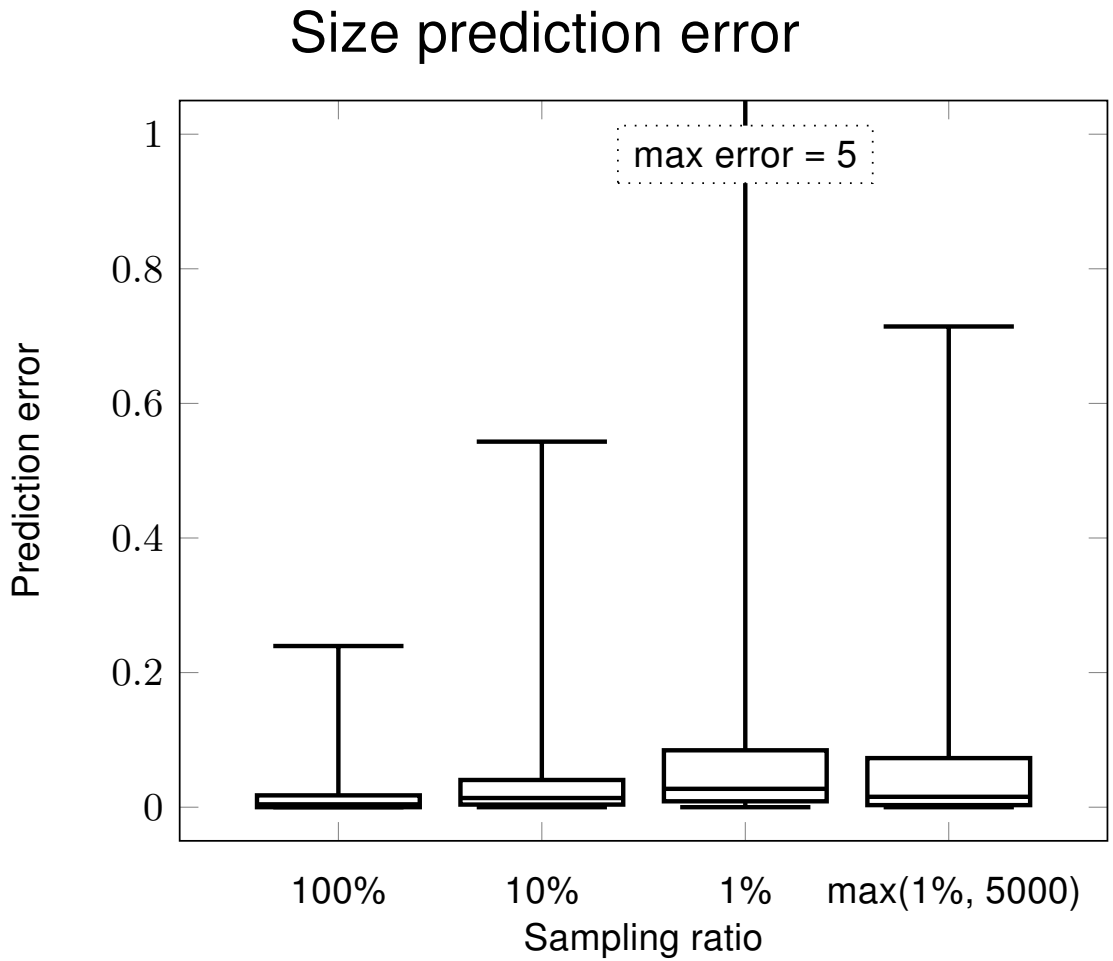
Example: array of Huffman encoded strings

- size $= |\text{data}| + \#\,\text{strings} \cdot |\text{pointer}|$
- data $= |\textit{raw data}| \cdot \textit{entropy}_0$

General idea: break down size into values that

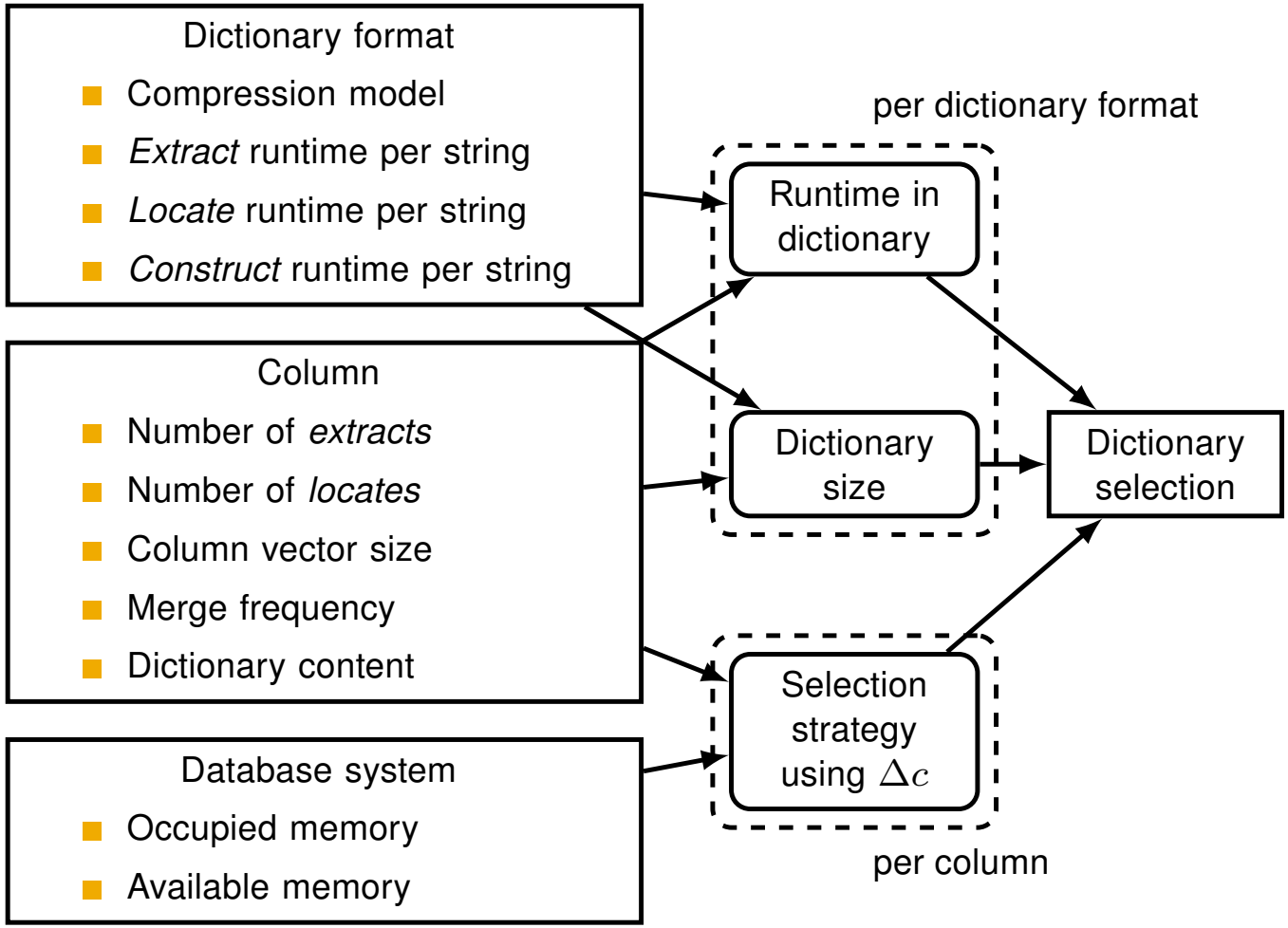
- are either **known** or
- can be **sampled**

Compression Models: Evaluation

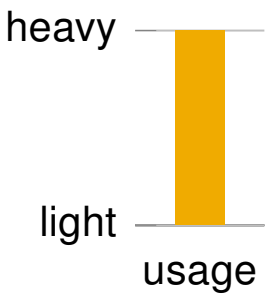


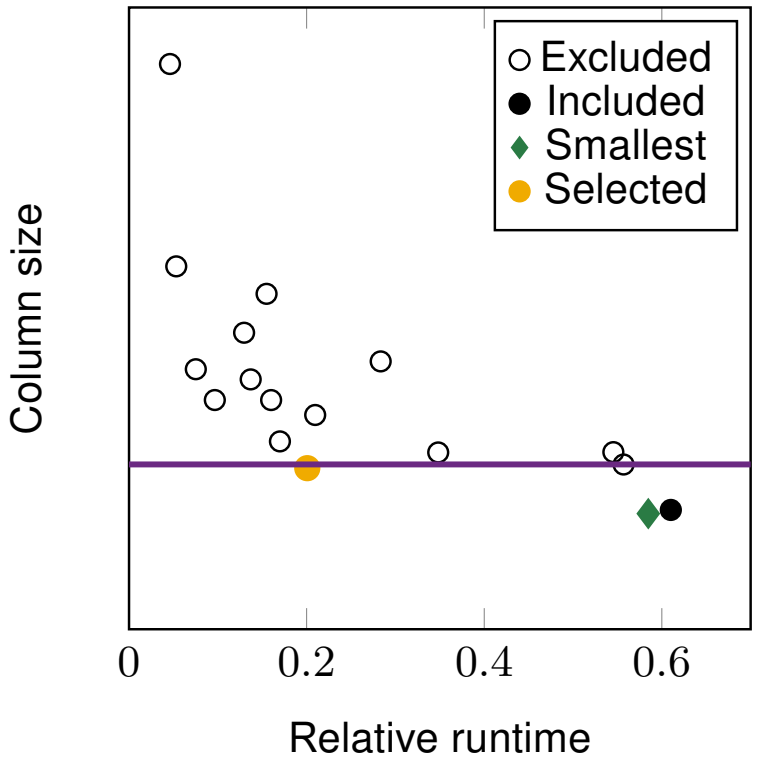
- Compression models offer cheap, but accurate enough size predictions

Automatic Dictionary Selection: Goals and Overview

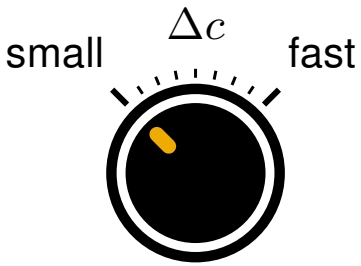


Trade-Off Selection Strategy



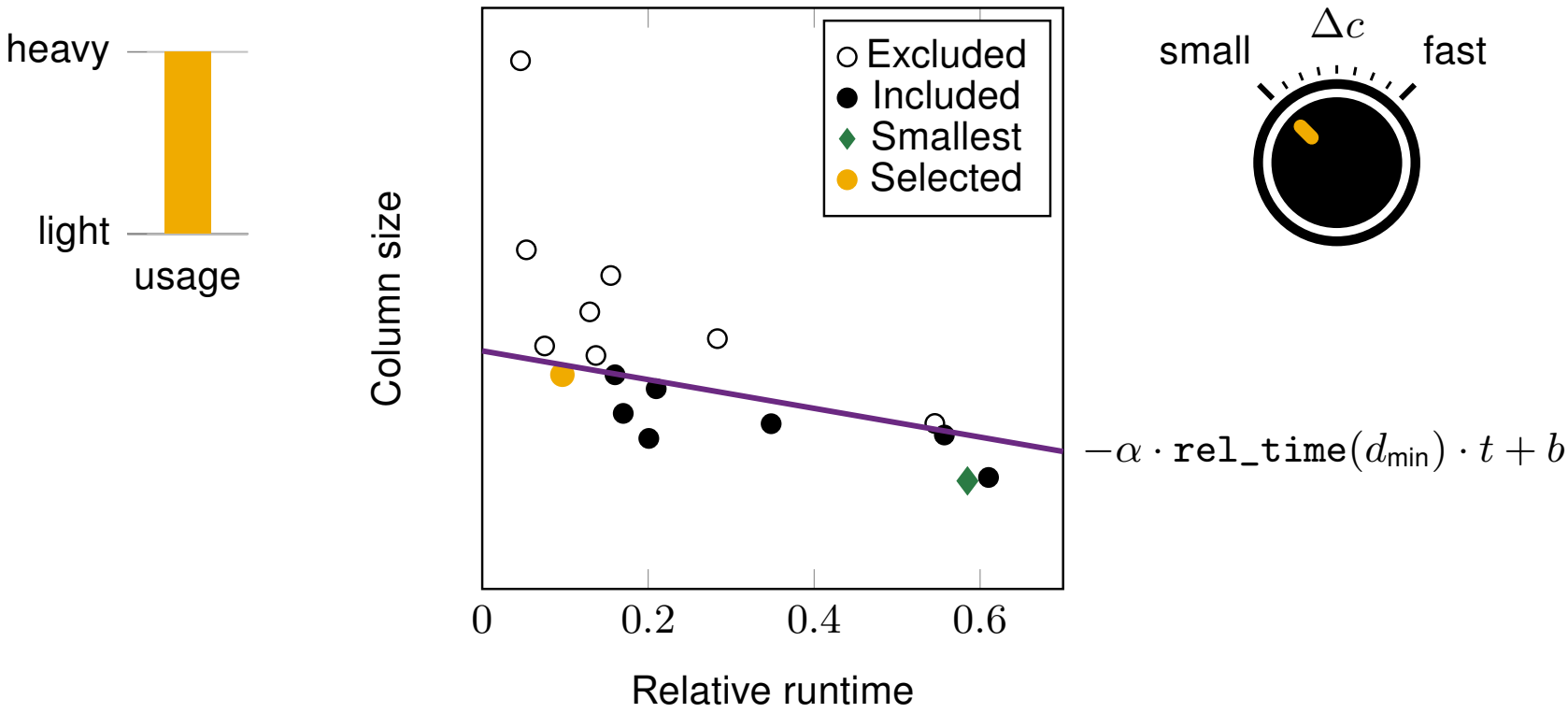


$(1 + \Delta c) \cdot \texttt{size}_{\text{min}}$ [Lem12]



- Our heuristic selects a dictionary format based on local information and a global trade-off parameter (Δc).

Trade-Off Selection Strategy

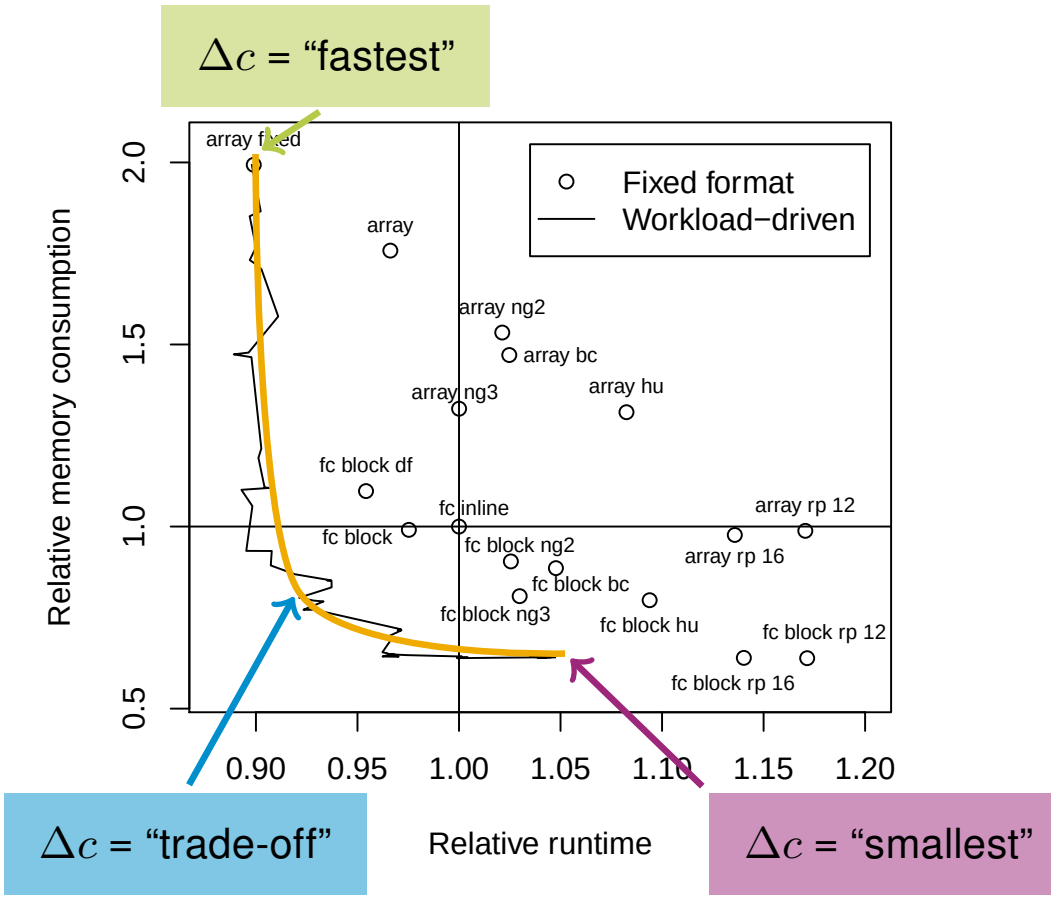


- Our heuristic selects a dictionary format based on local information and a global trade-off parameter (Δc).

Evaluation

Setup

- TPC-H with `*key` columns as strings ⇝ 47 dictionaries
- Workload: all queries consecutively



- Workload-driven dictionary selection outperforms static selection
- Δc effectively controls space / time trade-off

Summary

- Survey of dictionary implementations
  - Variety of space / time trade-offs depending on content
- Compression models
  - Feasible size prediction for assisted or automatic selection
- Automatic selection strategy
  - Heuristic translating a global trade-off parameter into local decisions
- Result
  - Workload-driven format selection improves overall system trade-off

Thank You!