Synthesis-Powered Visualization Authoring

https://falx.cs.washington.edu

Chenglong Wang

University of Washington

(In collaboration with: Yu Feng, Ras Bodik, Isil Dillig, Alvin Cheung, Amy Ko)

PAUL G. ALLEN SCHOOL
OF COMPUTER SCIENCE & ENGINEERING

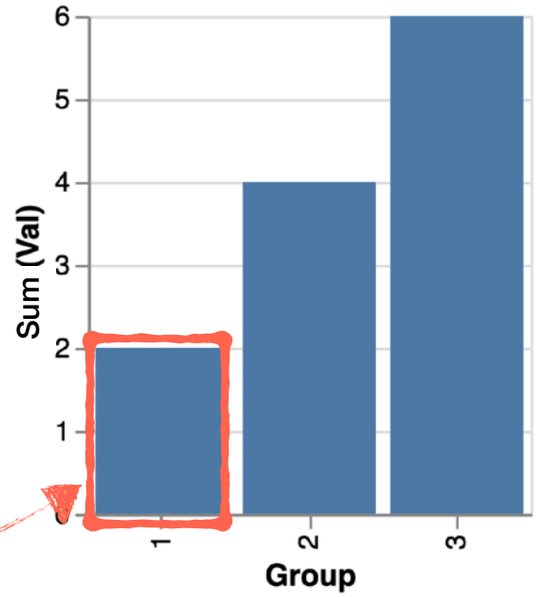
What is visualization?

"The grammar of graphics"

Visualization = mappings from *data columns* to *geometric properties*

Group	Val
1	2
2	4
3	6

bar(***x*** ← ***Group*****,**
height ← ***Val*** *)*

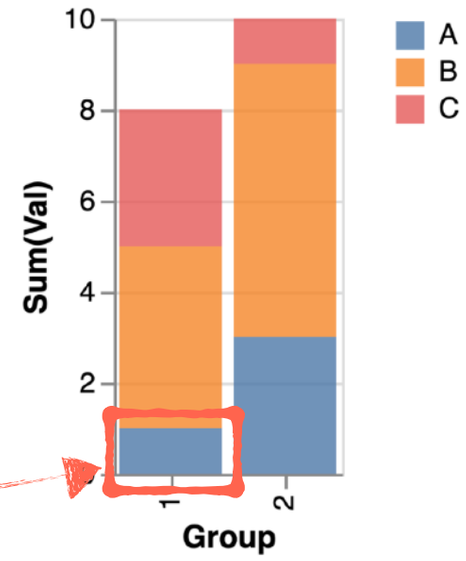


Wilkinson, Leland. "The grammar of graphics." *Handbook of Computational Statistics*. Springer, Berlin, Heidelberg, 2012. 375-414.

Task: Sum up values for each group in each column and map to a stacked bar chart

Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

?



The visualization expects an input data with three columns

Each maps to *x*, *height*, *color*

In practice: Visualization = data transformation + design specification

Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

gather(T, Group, [A,B,C])

Group	Key	Val
1	A	0
1	A	1
1	B	2
1	B	2
…	…	…

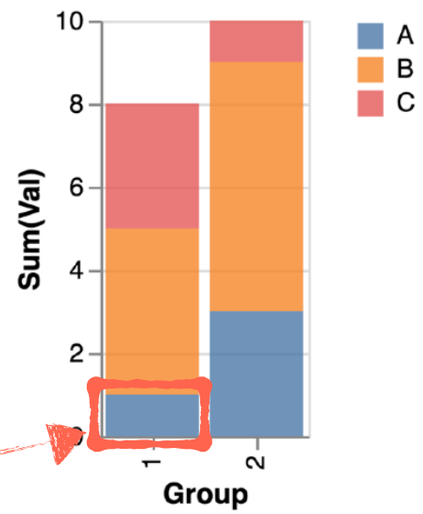
Gather
"transpose" the table

aggregate(
t,
group=[Group, Key],
target=sum(Val))

Aggregate
Summing up values for each group

Group	Key	Val
1	A	1
1	B	4
…	…	…

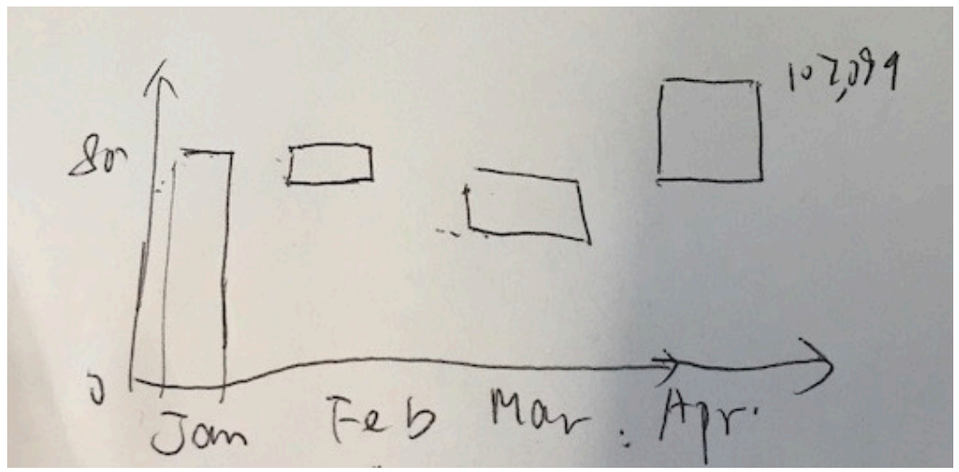
bar(x ← Group,
color ← Key
height ← Val)

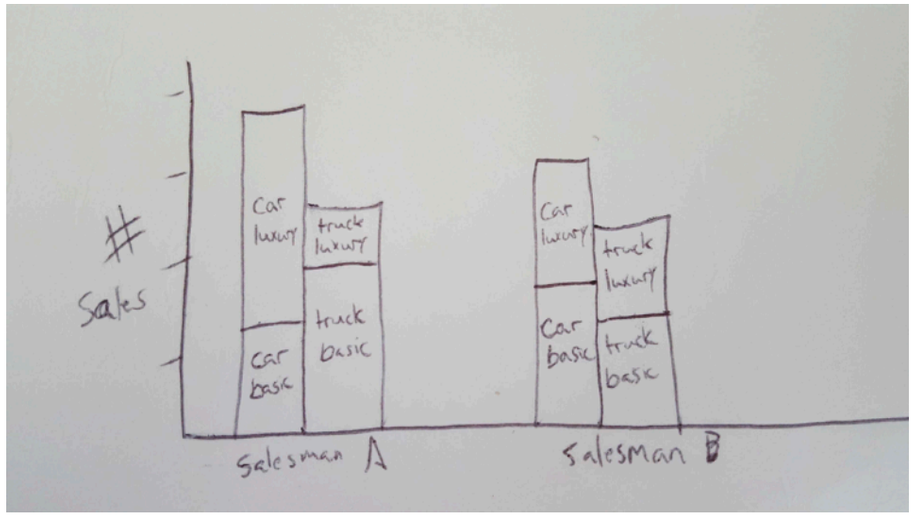


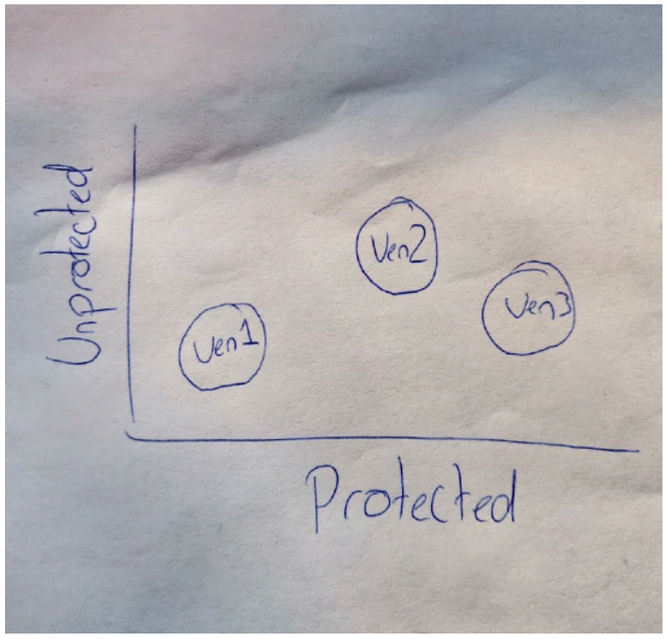
Challenge: Data transformation
Prepares the input data to match the visualization design.

[1] Gatto, Malu AC. "Making research useful: Current challenges and good practices in data visualization." (2015).
[2] Perkel, Jeffrey M. "Data visualization tools drive interactivity and reproducibility in online publishing." Nature 554.7690 (2018): 133-134.
[3] Kasica et al. "Table Scraps: An Actionable Framework for Multi-Table Data Wrangling From An Artifact Study of Computational Journalism" InfoVis (2020)

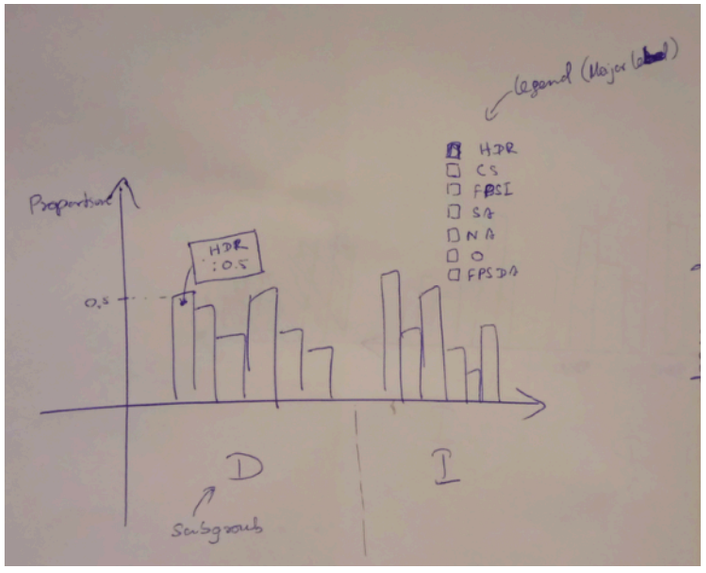
How end users communicate visualization ideas in practice?

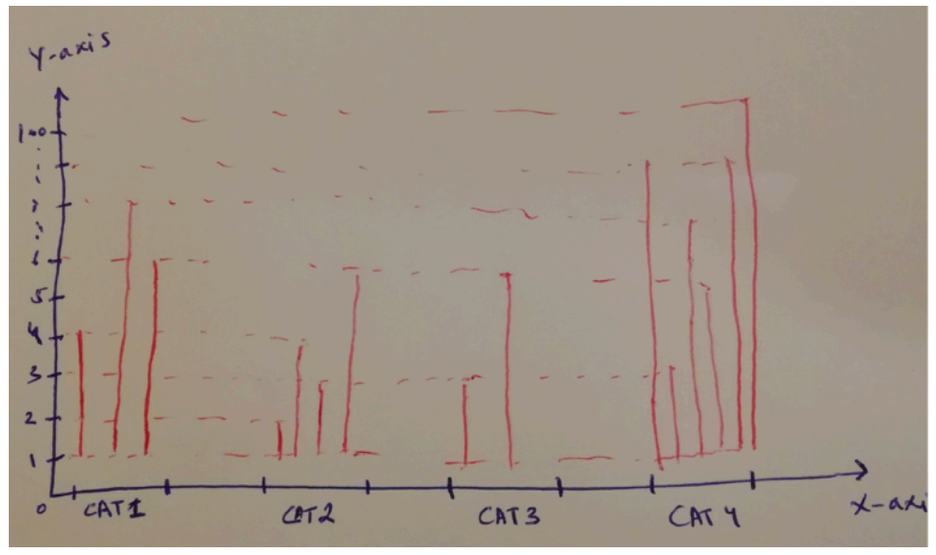






stackoverflow



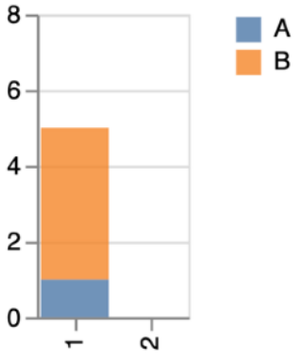


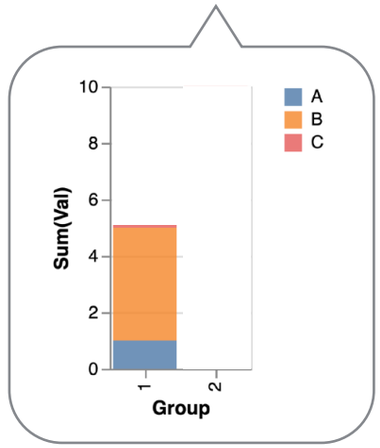
Observation: End users use partial visualizations to communicate their ideas.

Our design: Visualization by Example

Group	Key	Val
1	A	0
1	A	1
1	B	2
1	B	2
…	…	…

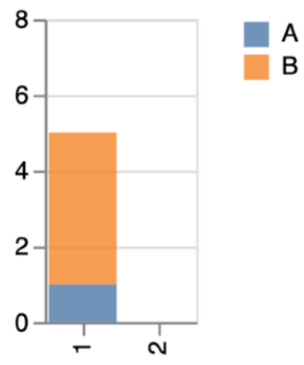
bar_1(x←1, h←1, color←A)
bar_2(x←1, h←4, color←B)





Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

bar_1(x←1, h←1, color←A)
bar_2(x←1, h←4, color←B)

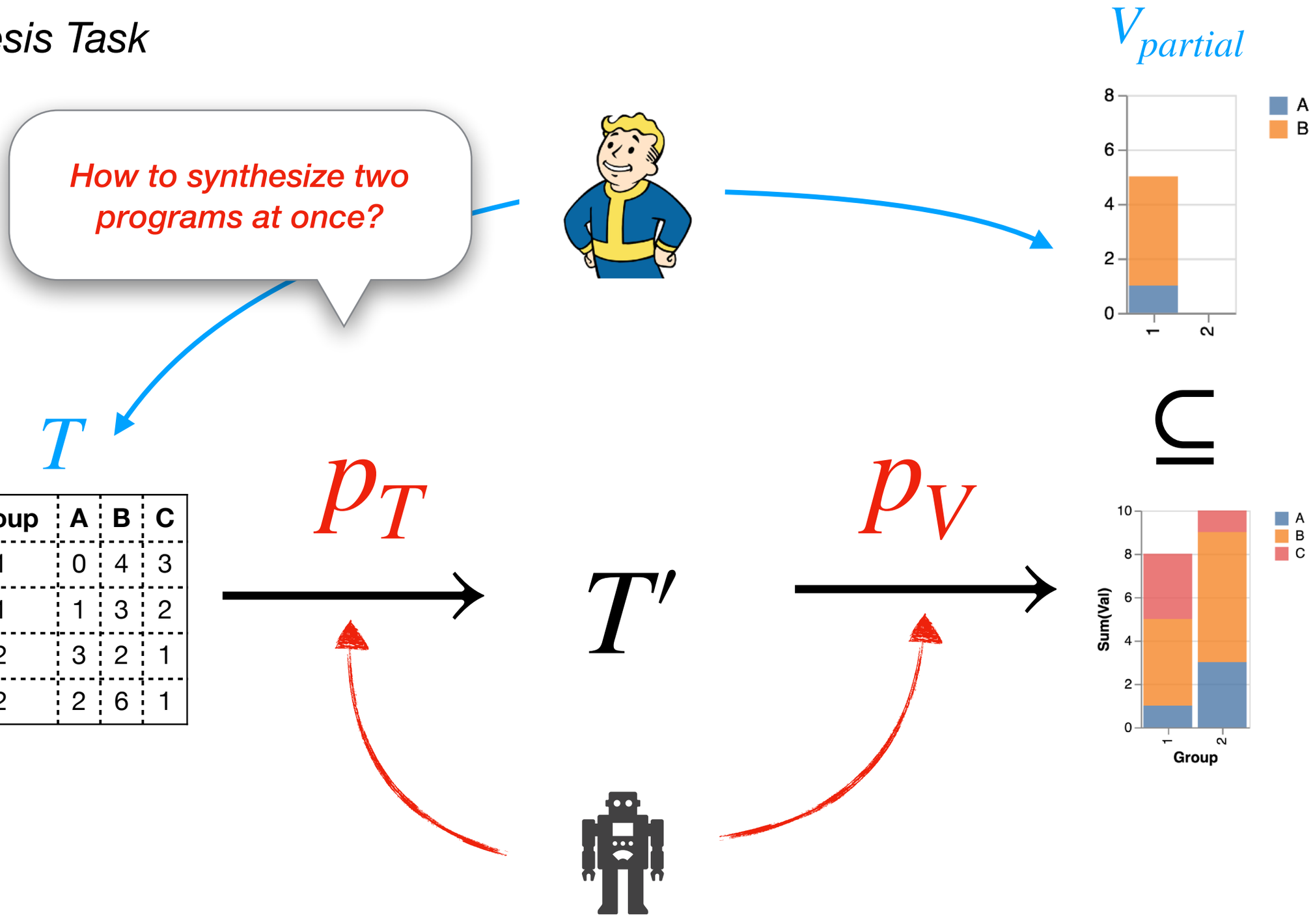


How can we synthesize visualization from examples?

Easy to specify

Data-layout independent

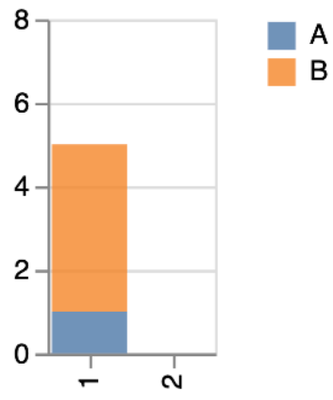
Synthesis Task



Synthesize p_T, p_V *such that the synthesized visualization* $p_V(p_T(\ T\ ))$ *contains* $V_{partial}$

Task decomposition

$V_{partial}$



T

Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

$\xrightarrow{p_T}$ T' $\xrightarrow{p_V}$ V

$\subseteq$

Task decomposition

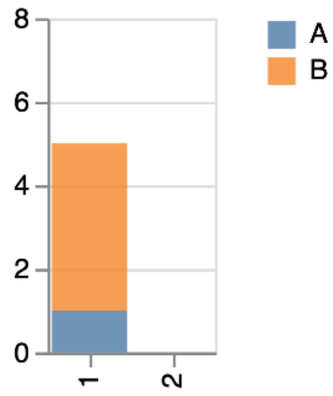
Task 1: Decompile visualization

$p_V(T^\star) = V_{partial}$

Task 2: Synthesize data transformation

Objective: $p_T(T) \supseteq T^\star$

$V_{partial}$



T

Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

p_T

$T^\star$

C1	C2	C3
1	A	1
1	B	4

p_V

$\subseteq$

T'

bar(x ← C1,
height ← C3,
color ← C2)

$\subseteq$

V

Step 2: Synthesize Data Transformation

T

Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

$T^\star$

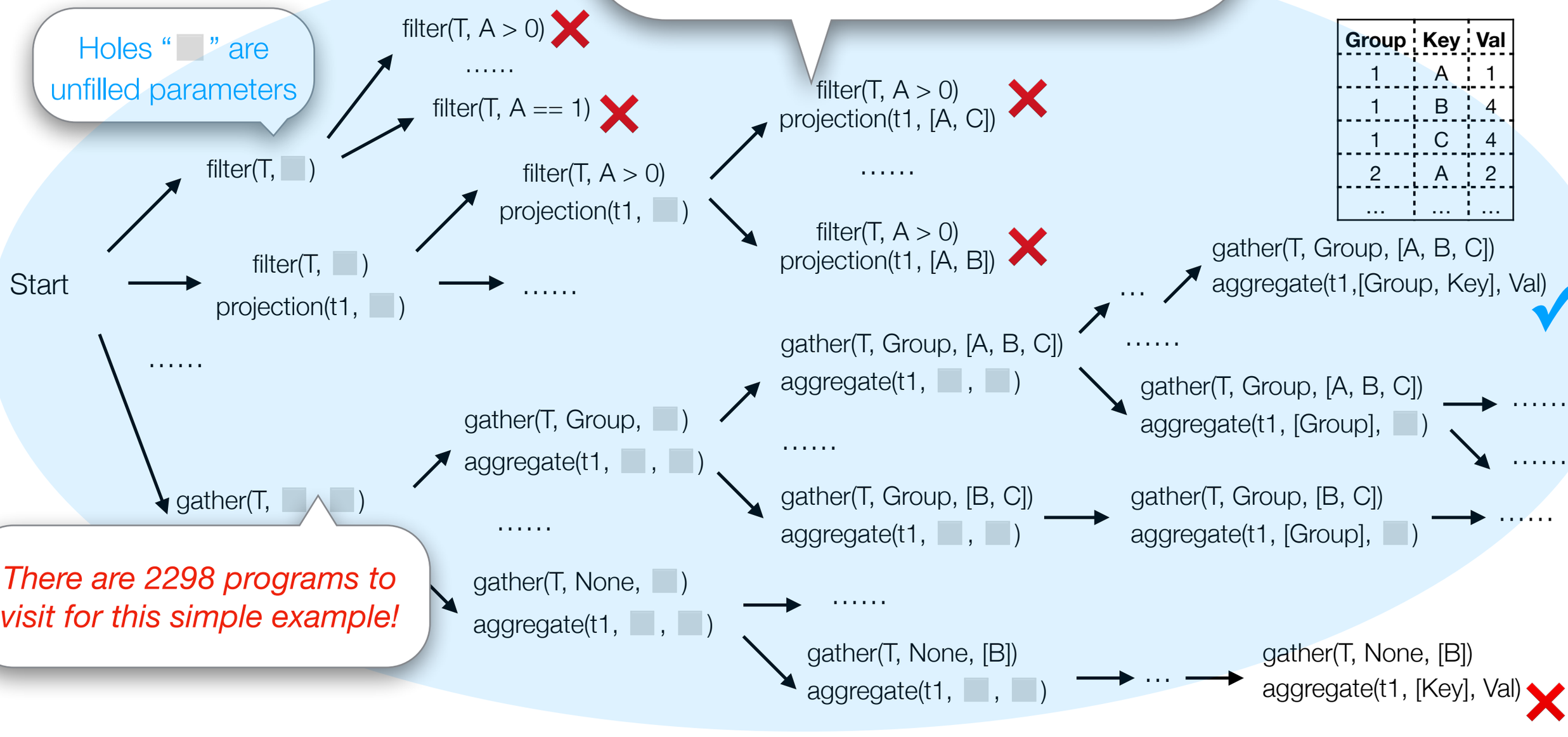
C1	C2	C3
1	A	1
1	B	4

Objective: synthesize p_T such that $T^\star \subseteq p_T(T)$

gather, spread, separate, te, join, filter, select, ggregate, …..

How can we make this combinatorial search process tractable?

Holes " " are unfilled parameters



Group	Key	Val
1	A	1
1	B	4
1	C	4
2	A	2
…	…	…

There are 2298 programs to visit for this simple example!

Step 2: Synthesize Data Transformation

T

Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

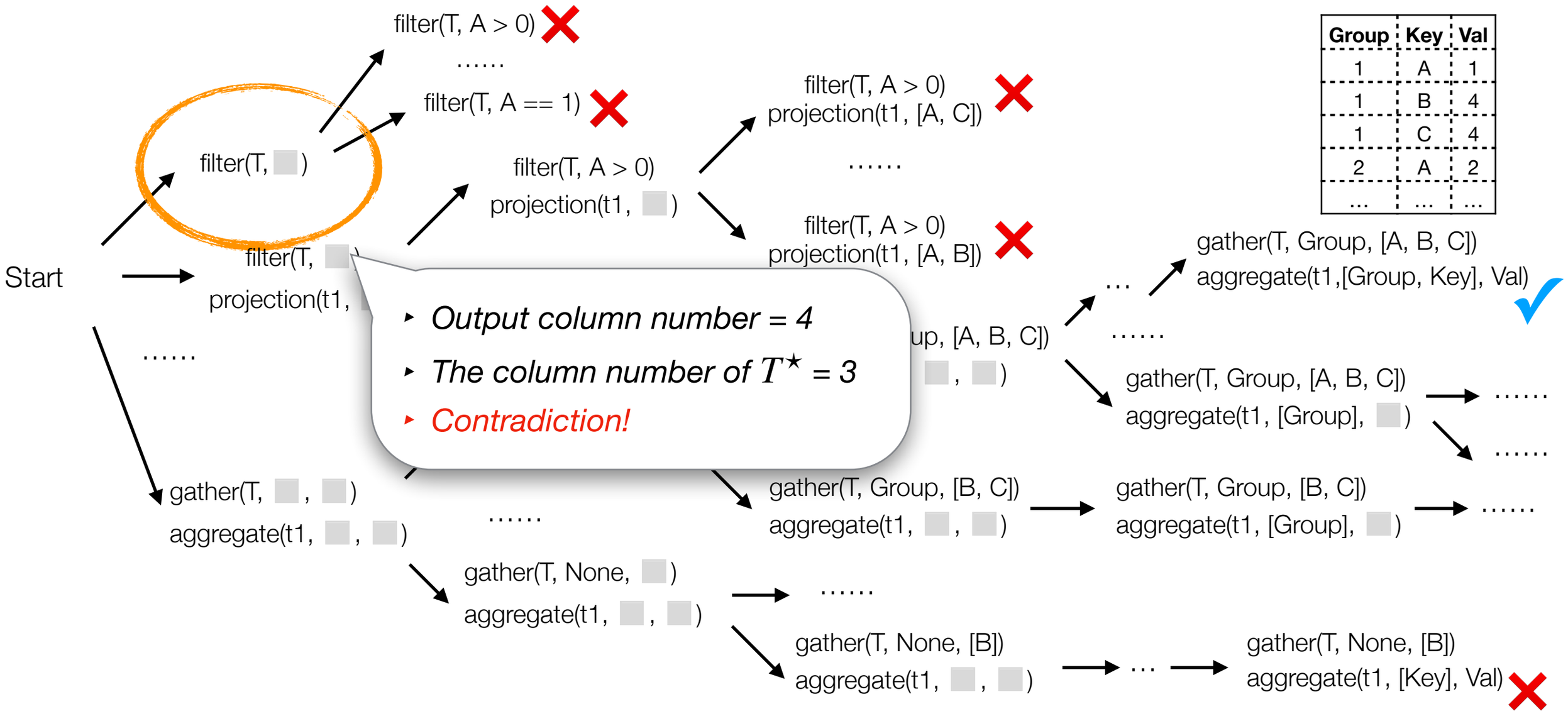
$T^\star$

C1	C2	C3
1	A	1
1	B	4

Objective: synthesize p_T such that $T^\star \subseteq p_T(T)$

Option: Deductive reasoning

Morpheus [PLDI'17, PLDI'18], Lens [POPL'20]



Group	Key	Val
1	A	1
1	B	4
1	C	4
2	A	2
...	...	...

Step 2: Synthesize Data Transformation

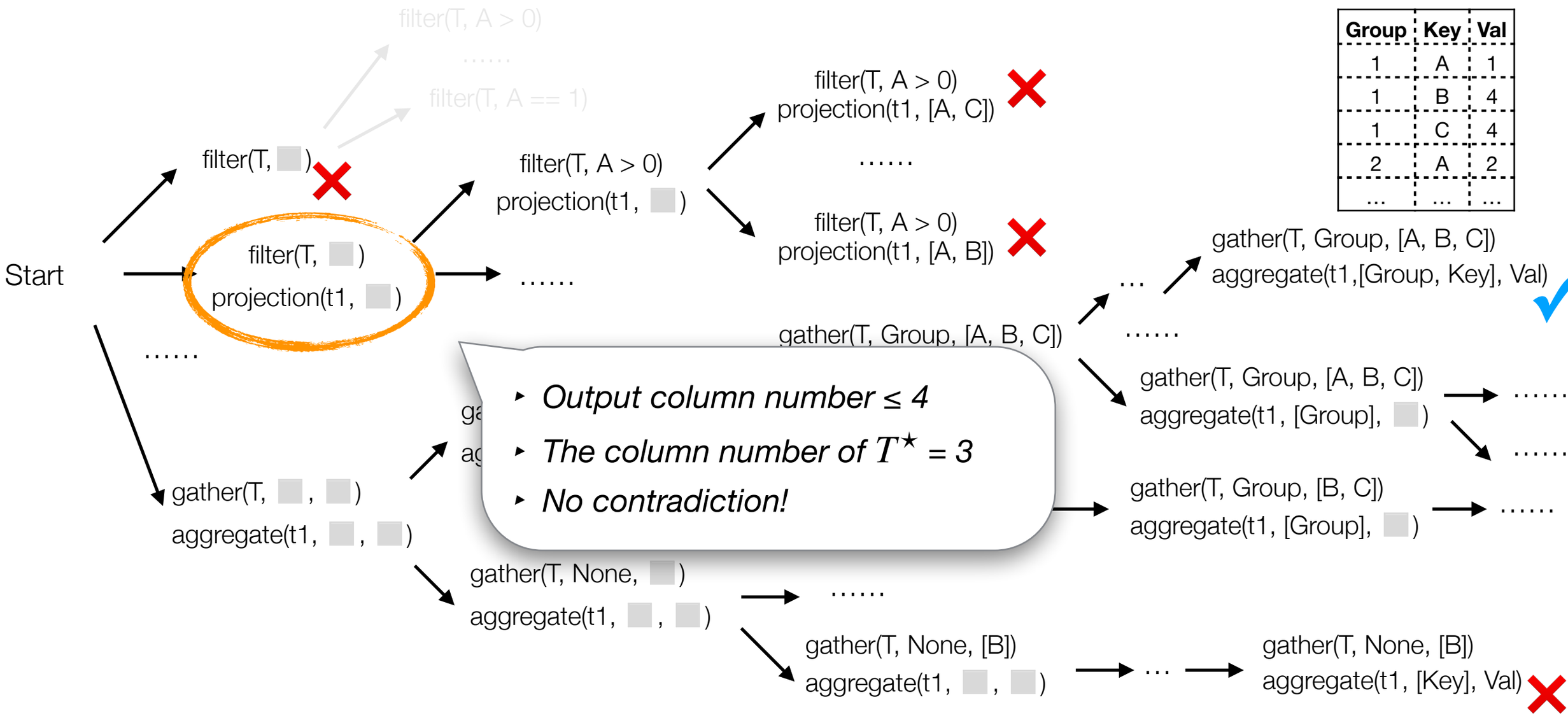
Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

C1	C2	C3
1	A	1
1	B	4

Objective: $T^{\star} \subseteq p_T(T)$

Option: Deductive reasoning

Morpheus [PLDI'17, PLDI'18], Lens [POPL'20]



Step 2: Synthesize Data Transformation

Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

C1	C2	C3
1	A	1
1	B	4

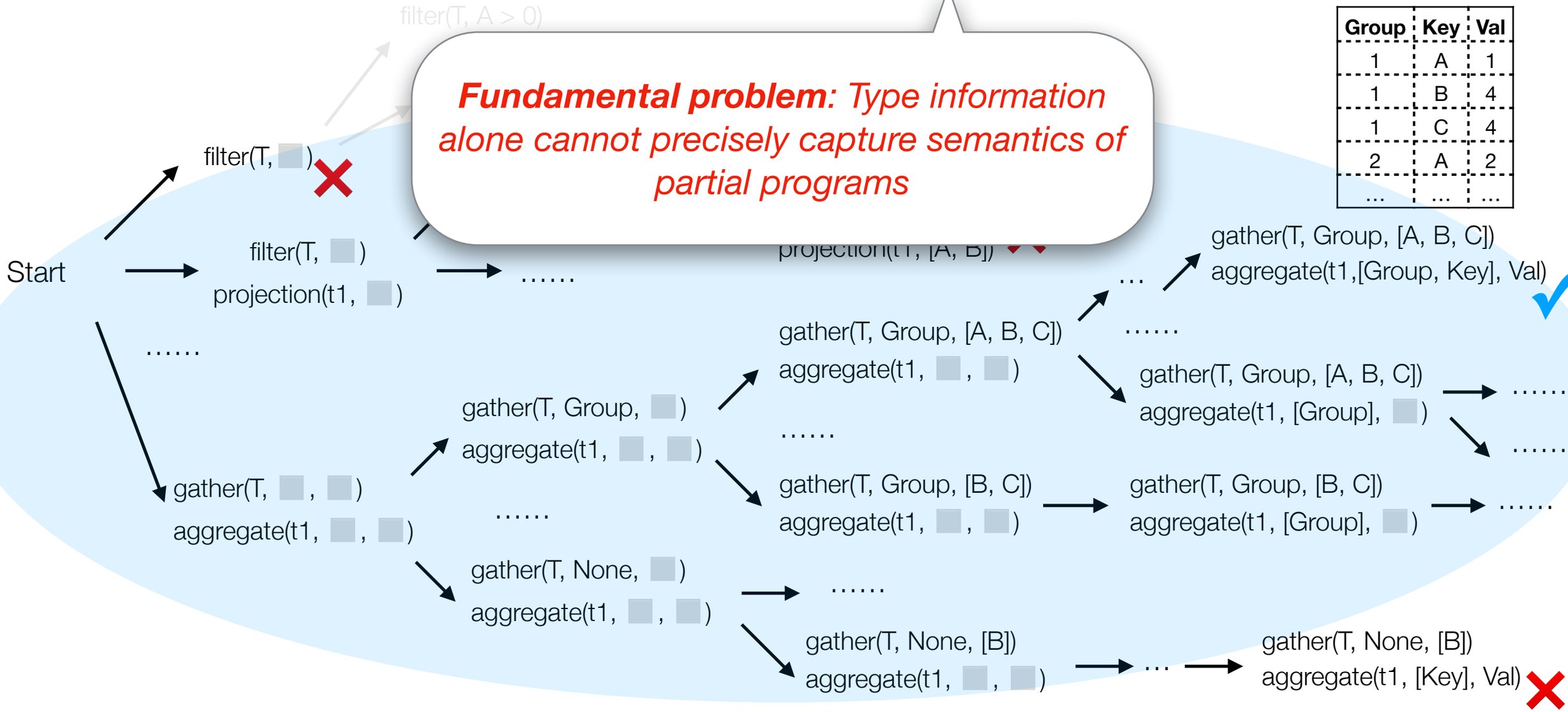
Objective: $T^{\star} \subseteq p_T(T)$

Option: Deductive reasoning

Morpheus [PLDI'17, PLDI'18], Lens [POPL'20]

***Fundamental problem**: Type information alone cannot precisely capture semantics of partial programs*

Group	Key	Val
1	A	1
1	B	4
1	C	4
2	A	2
...	...	...



Step 2: Synthesize Data Transformation

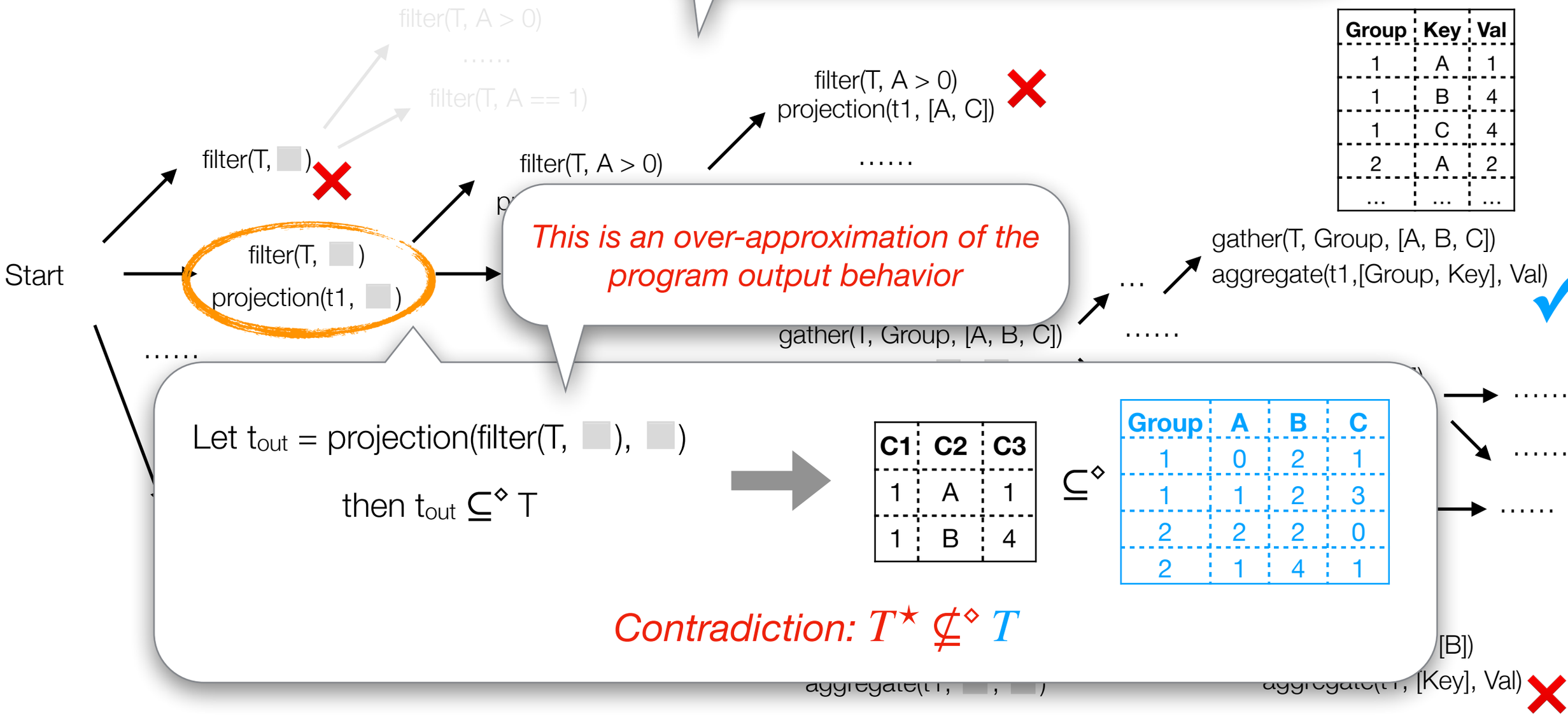
Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

C1	C2	C3
1	A	1
1	B	4

Objective: $T^{\star} \subseteq p_T(T)$

Reasoning with abstract semantics

Propagate concrete values through partial programs to analyze their behaviors



Step 2: Synthesize Data Transformation

Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

C1	C2	C3
1	A	1
1	B	4

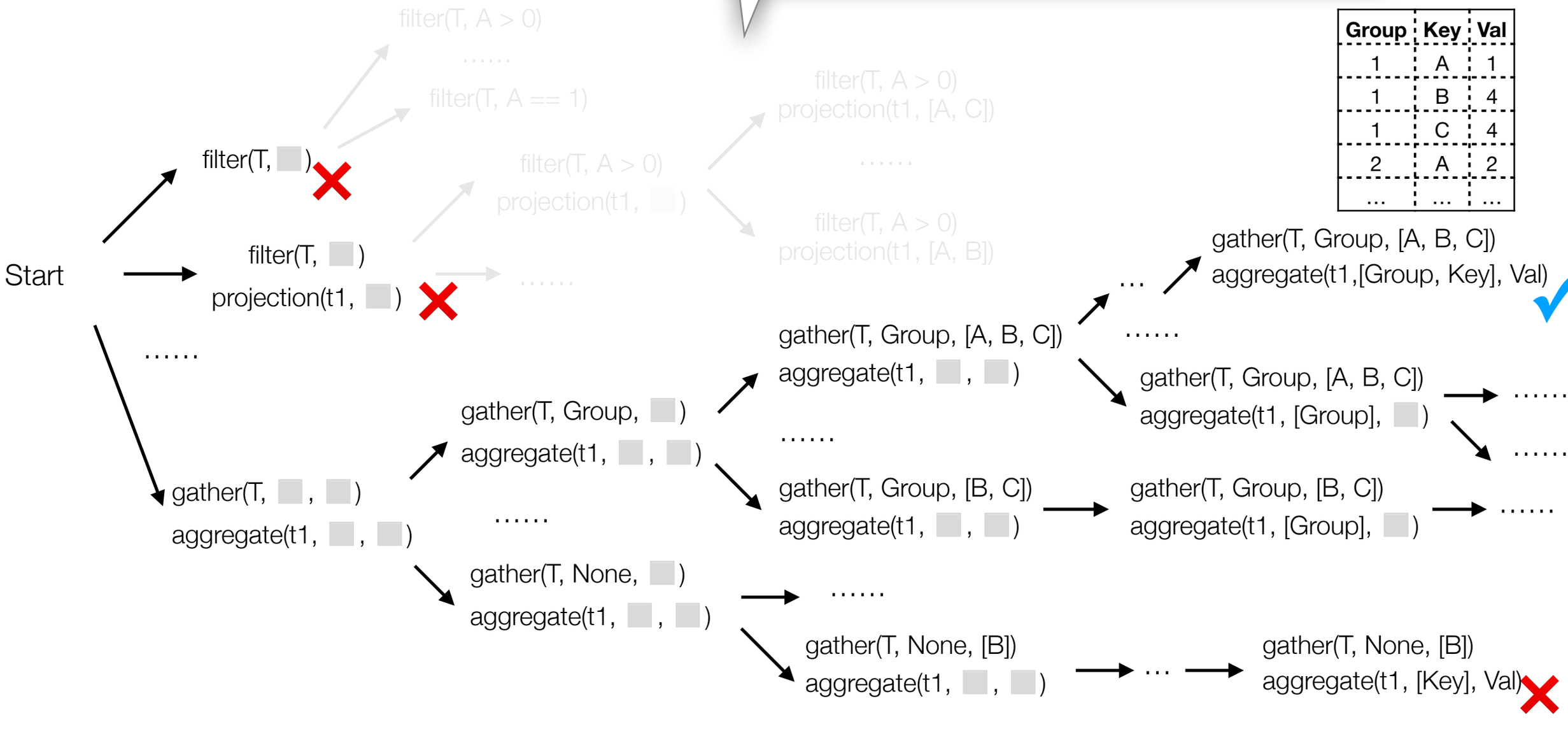
Objective: $T^{\star} \subseteq p_T(T)$

Reasoning with abstract semantics

Propagate concrete values through partial programs to analyze their behaviors

[PLDI 17, POPL 20]

Group	Key	Val
1	A	1
1	B	4
1	C	4
2	A	2
...	...	...



Step 2: Synthesize Data Transformation

Group	A	B	C
1	0	2	1
1	1	2	3
2	2	2	0
2	1	4	1

C1	C2	C3
1	A	1
1	B	4

Objective: $T^{\star} \subseteq p_T(T)$

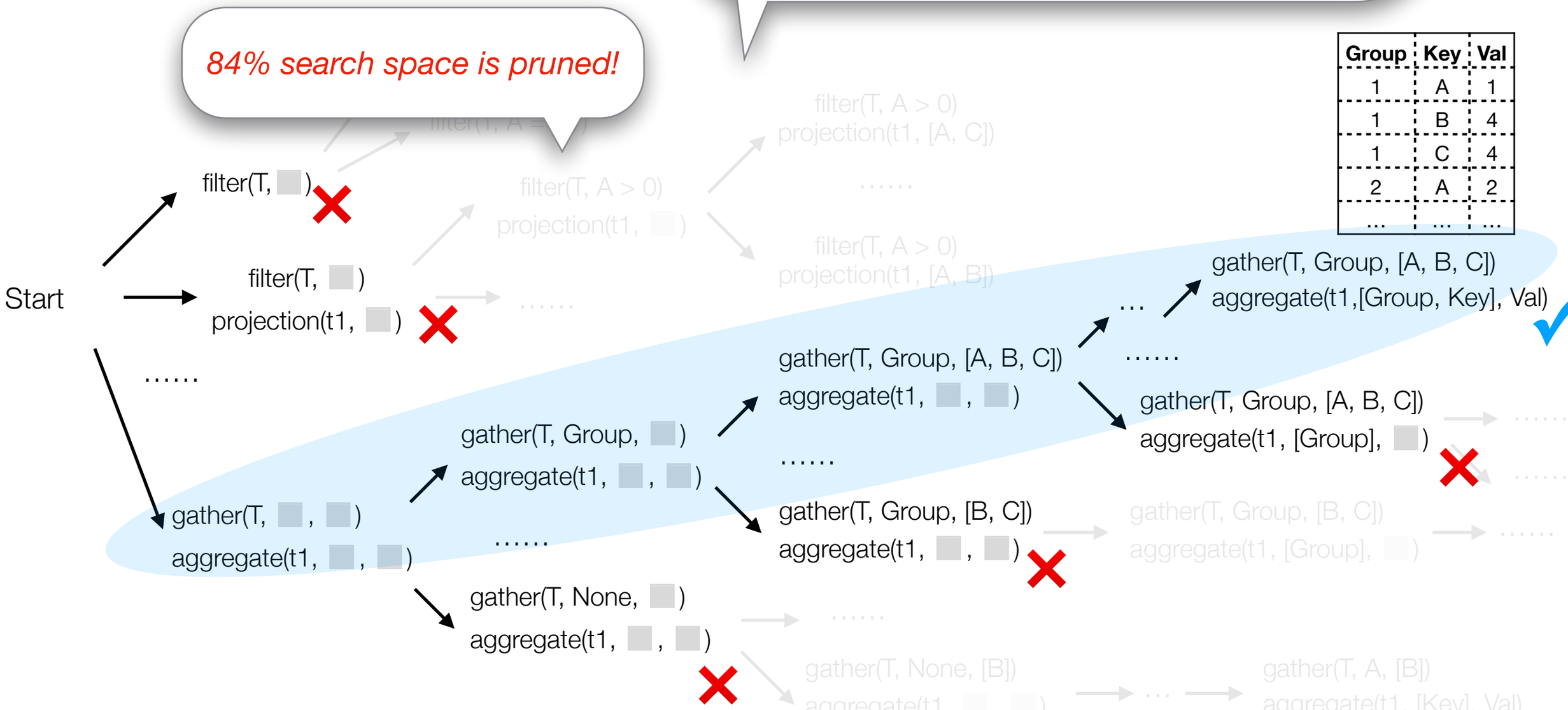
Reasoning with abstract semantics

Propagate concrete values through partial programs to analyze their behaviors

[PLDI 17, POPL 20]

84% search space is pruned!

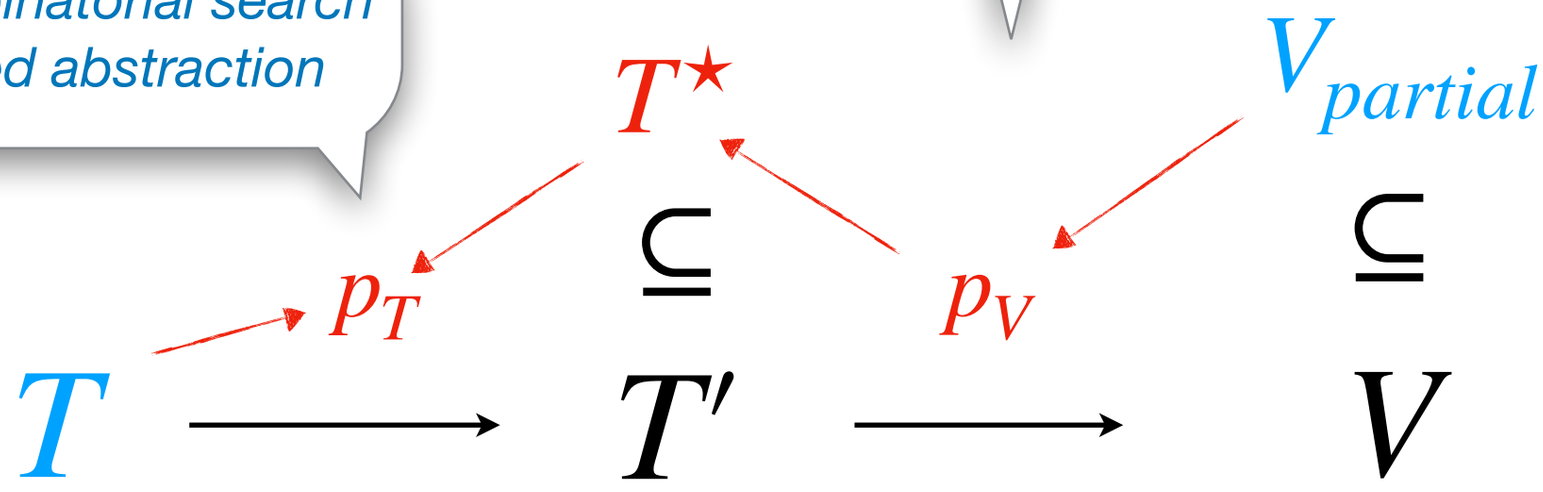
Group	Key	Val
1	A	1
1	B	4
1	C	4
2	A	2
...	...	...



Synthesis algorithm review

1. Decompose the task with visualization decompilation

2. Scale up combinatorial search with value-based abstraction



Language support

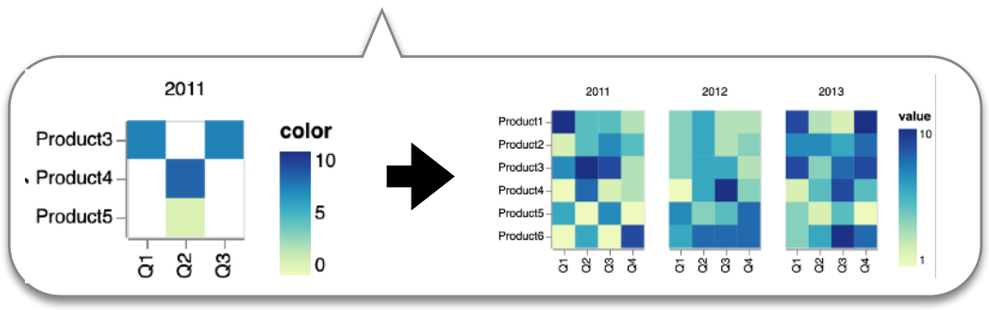
- R tidyverse
- Vega-Lite

83 benchmarks

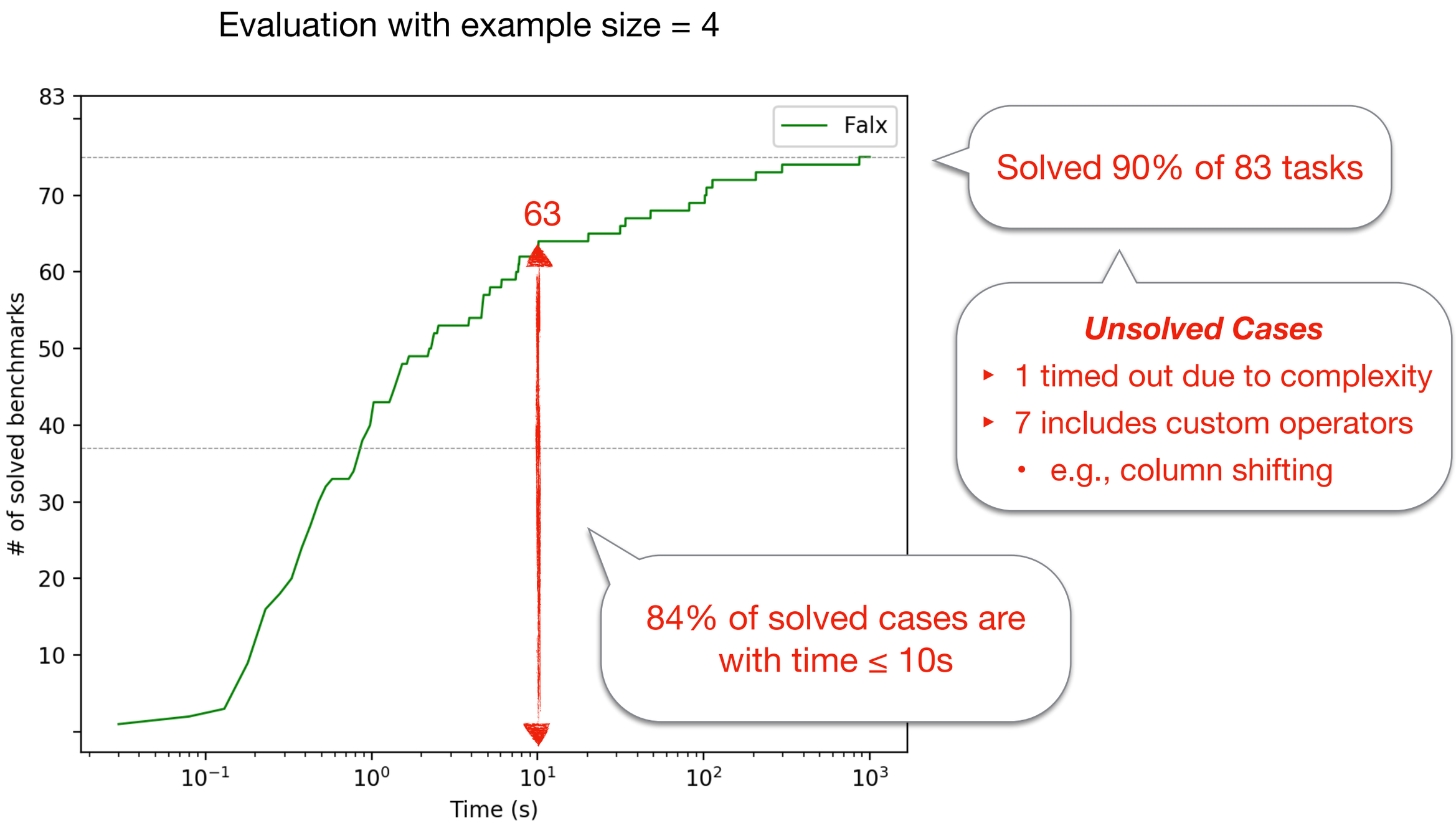
- Stack Overflow
- Excel / R tutorials

Process

- Test synthesis time with partial visualizations sampled from full visualizations.

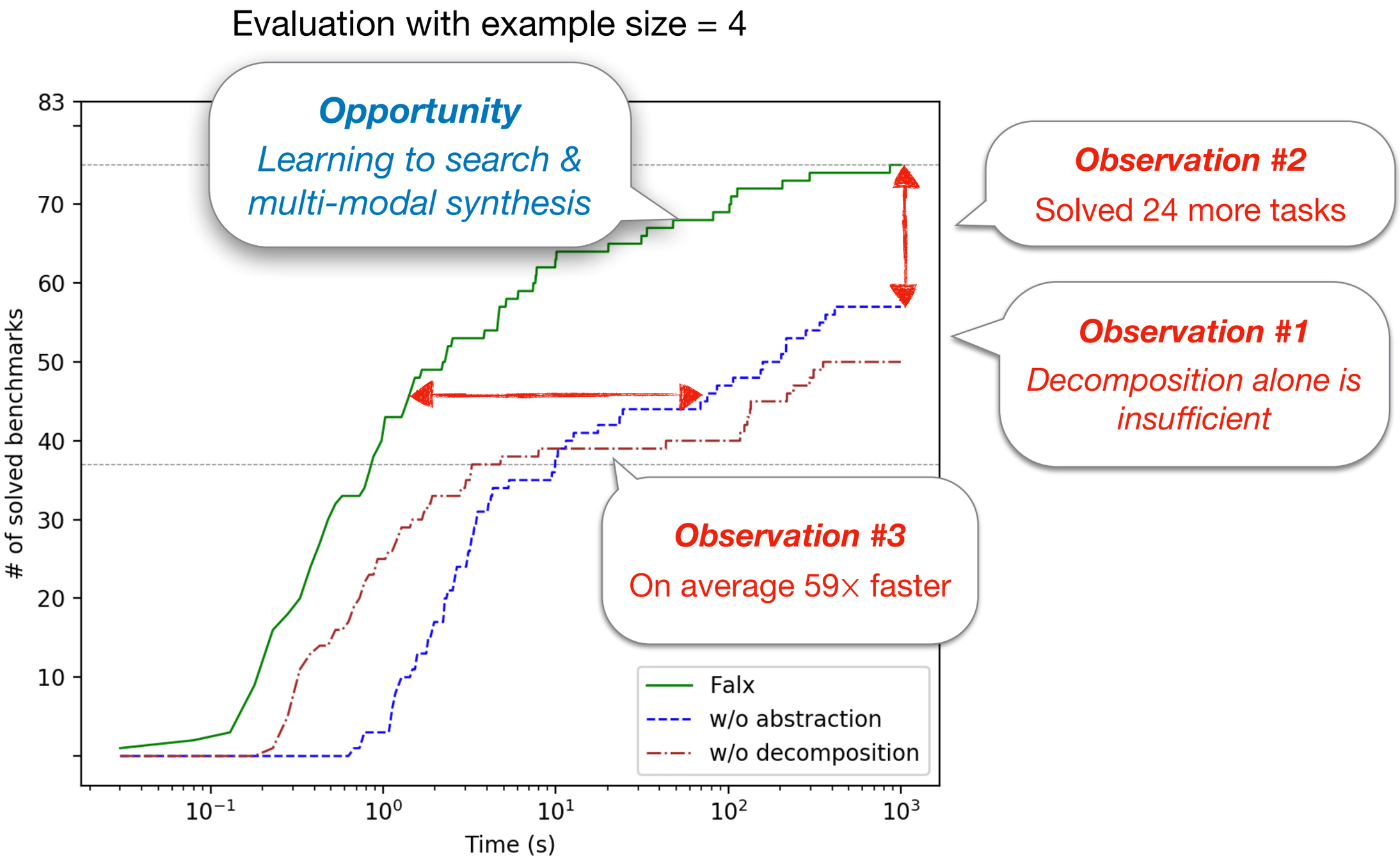


Can Falx solve practical visualization tasks?



Number of benchmarks that can be solved within given time limit (log scale)

Does abstraction and decomposition play important roles in Falx?

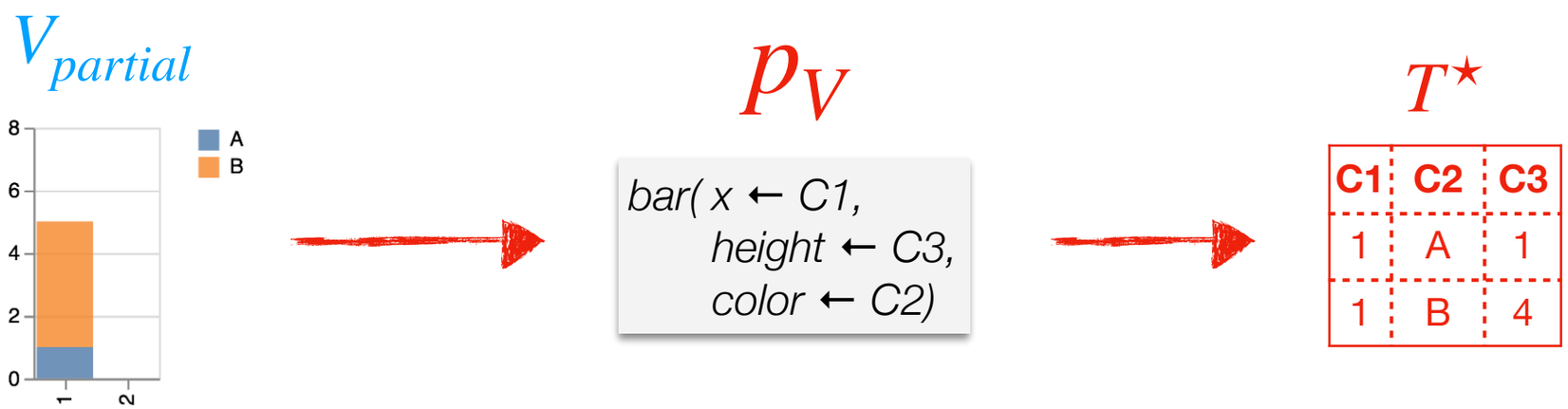


Number of benchmarks that can be solved within given time limit (log scale)

What I didn't cover:

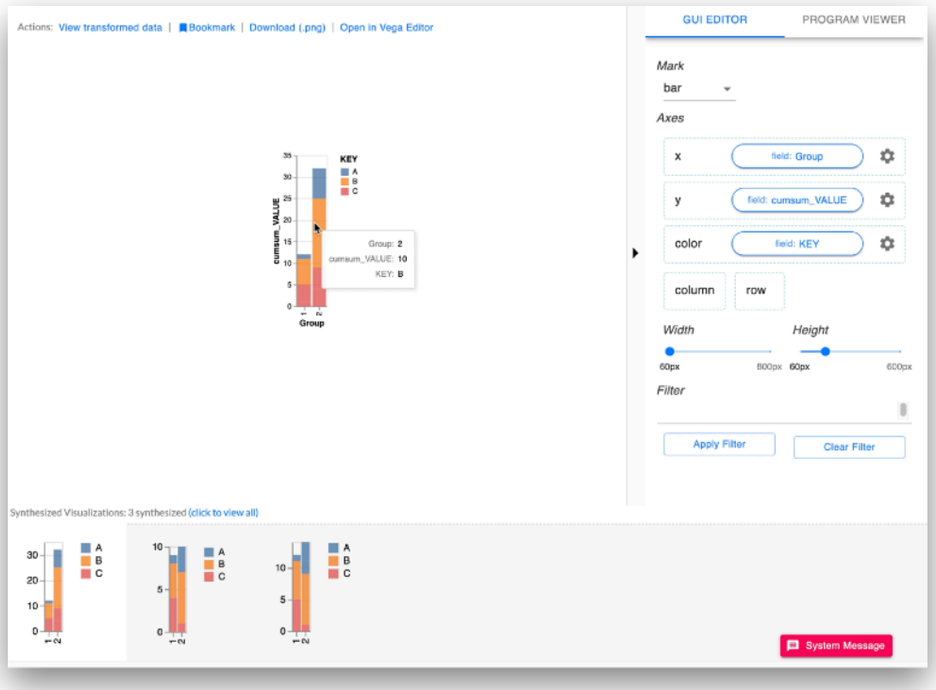
Visualization by Example [POPL 20]

(1) How to design the language $\mathscr{L}_V$ to support easy decompilation



(2) Interactive model design

(e.g., how to create example / how to handle ambiguity)



Falx: Synthesis-powered Visualization Authoring [CHI 21 Best Paper]

Visit falx.cs.washington.edu *now!*