Visualizations to Summarize Search Behavior

Ian S. Howell [1,2], Berthe Y. Choueiry [1], and Hongfeng Yu [2]

Constraint Systems Laboratory [1]

Visualizations Laboratory [2]

Department of Computer Science & Engineering

University of Nebraska-Lincoln

Introduction

- Backtrack search when solving a constraint satisfaction problem (CSP) suffers from thrashing
- Enforcing a consistency property reduces thrashing at the cost of further processing
- The tradeoff of running a higher-level consistency (HLC) is poorly understood.
- We propose to summarize search

Background

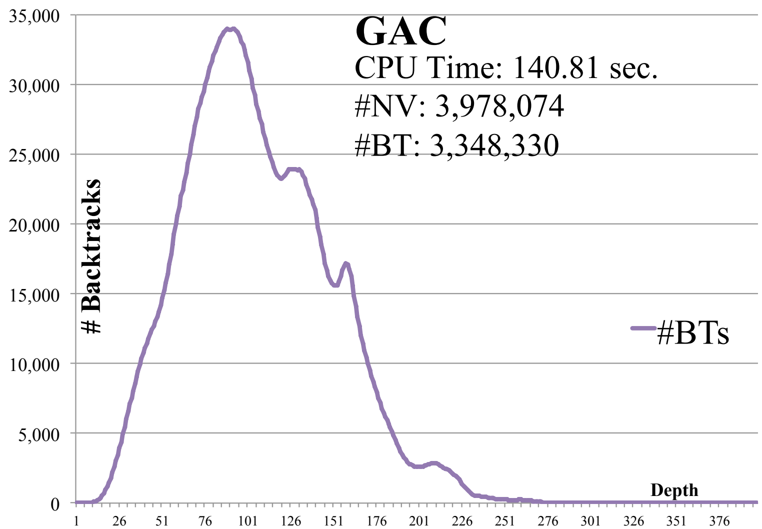
- A constraint satisfaction problem (CSP) consists of a set of variables, the variables' domains, and constraints.
- A solution is a domain to variable assignment that satisfies all constraints.
- Backtrack search is only sound and complete method of solving a CSP.

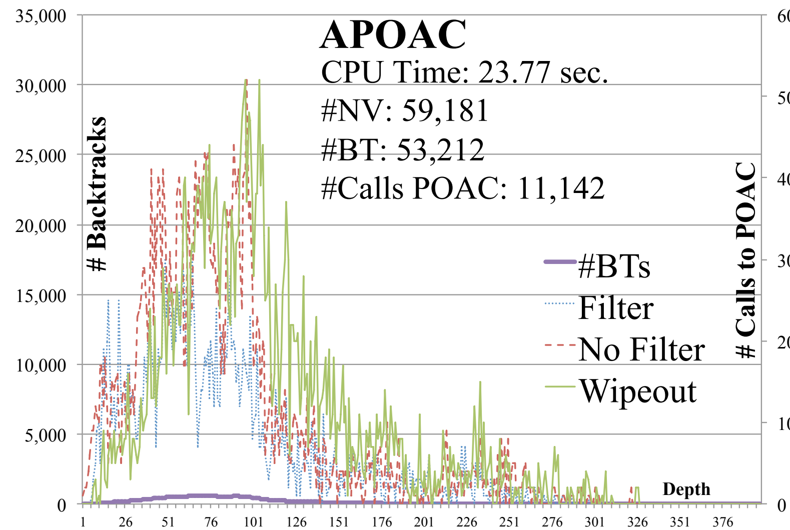
Background

- Thrashing is the main malady of search
  - Repeatedly performing same logic
- Applying a consistency algorithm after variable instantiation reduces thrashing
- Past visualizations focus on debugging and inspecting search
  - Detect thrashing through isomorphic subtrees
  - Debug individual issues by investigating the state of variables, propagators, etc.

Background

- Woodward et al. propose to measure:
  - thrashing with Backtracks per Depth (BpD)
  - HLC cost with consistency calls per depth (CpD)





Contributions

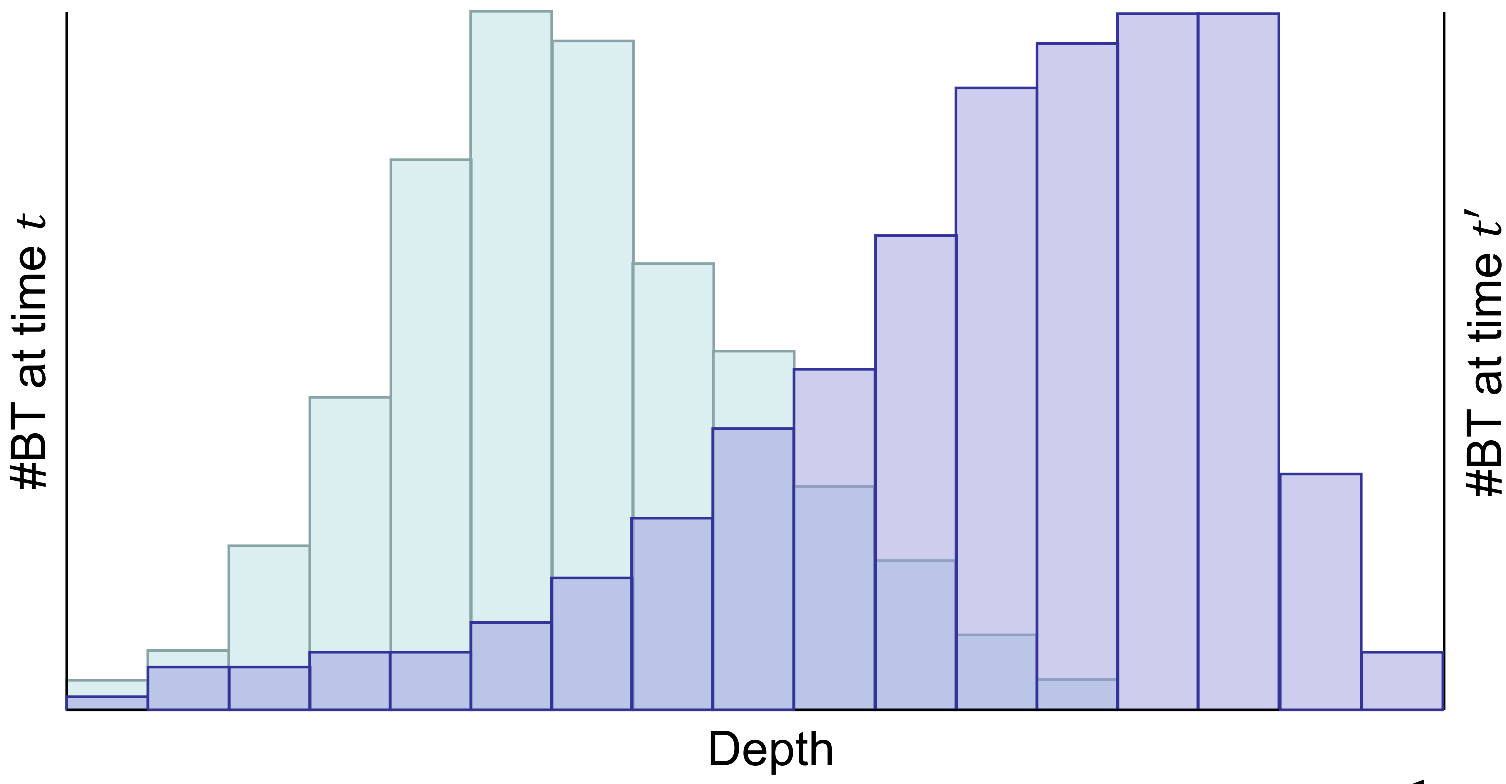
1. Criteria for computing distance between two time samples based on the BpD
2. A clustering technique for summarizing search into a history of qualitatively distinct regimes
3. A new visualization that examines the behavior of variable ordering heuristics

Outline

1. Introduction
2. Background
3. Contributions
4. **Outline**
5. Distance between timestamps
6. Summarizing search history
7. Example: Analyzing structure
8. Visualizing variable ordering
9. Example: Variable ordering
10. Conclusions and discussion

Distance between timestamps

- Design goal: capture the "shape" change



Distance between timestamps

- Adapt Kulbak-Liebler Divergence
- BpD distribution:
  - Use additive smoothing to account for zero-probability intervals

$$\widehat{p}(d,t) = \frac{\mathrm{BT}(d,t) + \alpha}{\sum_{d \in D} \mathrm{BT}(d,t) + \alpha d_{\max}}$$

Distance between timestamps

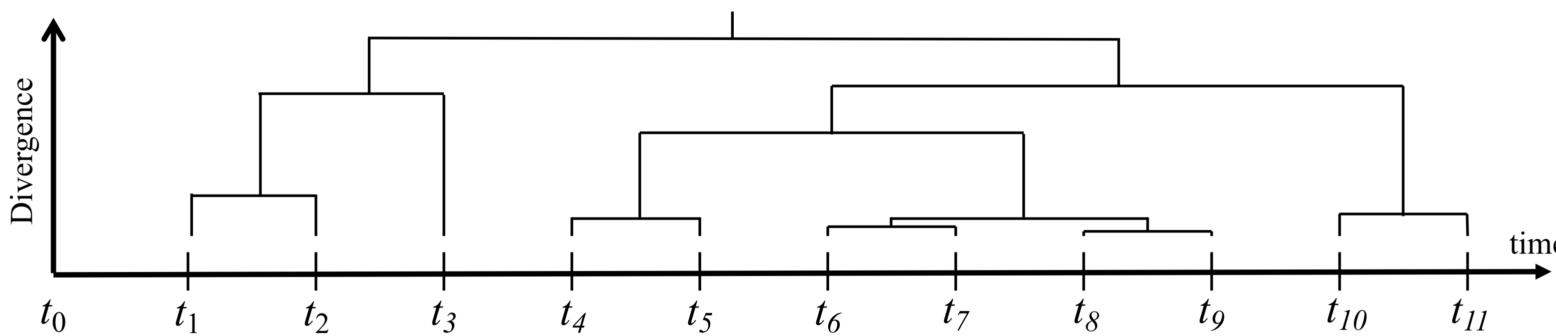
$$\mathrm{div}(\mathrm{t},\mathrm{t}') = \max\left(\sum_{d\in D}\widehat{p}(d,t)\log\left(\frac{\widehat{p}(d,t)}{\widehat{p}(d,t')}\right), \sum_{d\in D}\widehat{p}(d,t')\log\left(\frac{\widehat{p}(d,t')}{\widehat{p}(d,t)}\right)\right)$$

$KL(t,t')$ → $\sum_{d\in D}\widehat{p}(d,t)\log\left(\frac{\widehat{p}(d,t)}{\widehat{p}(d,t')}\right)$

$KL(t',t)$ → $\sum_{d\in D}\widehat{p}(d,t')\log\left(\frac{\widehat{p}(d,t')}{\widehat{p}(d,t)}\right)$

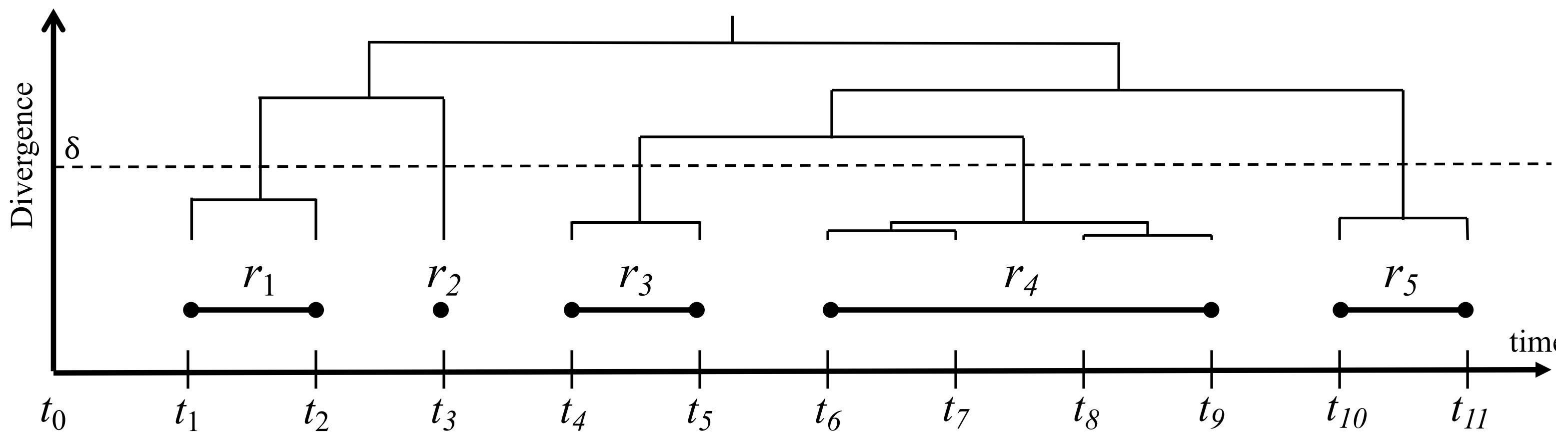
Summarizing search history

- Create a clustering tree
  - Use agglomerative, hierarchical clustering
  - Only merge temporally adjacent clusters
  - Each cluster's representative is the middle of the interval of included timestamps

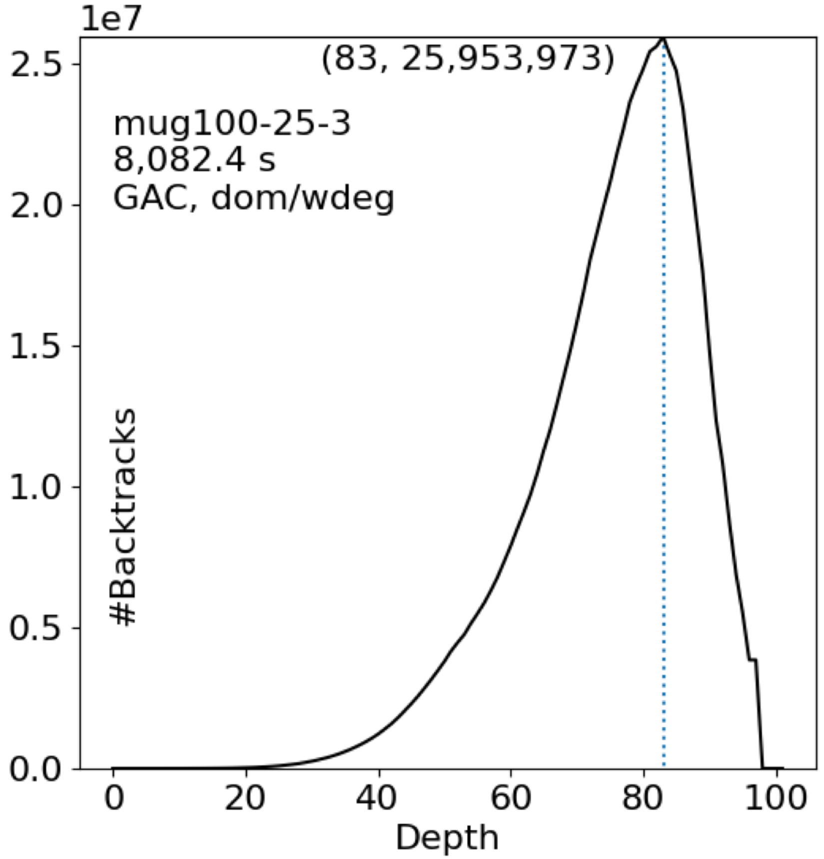


Summarizing search history

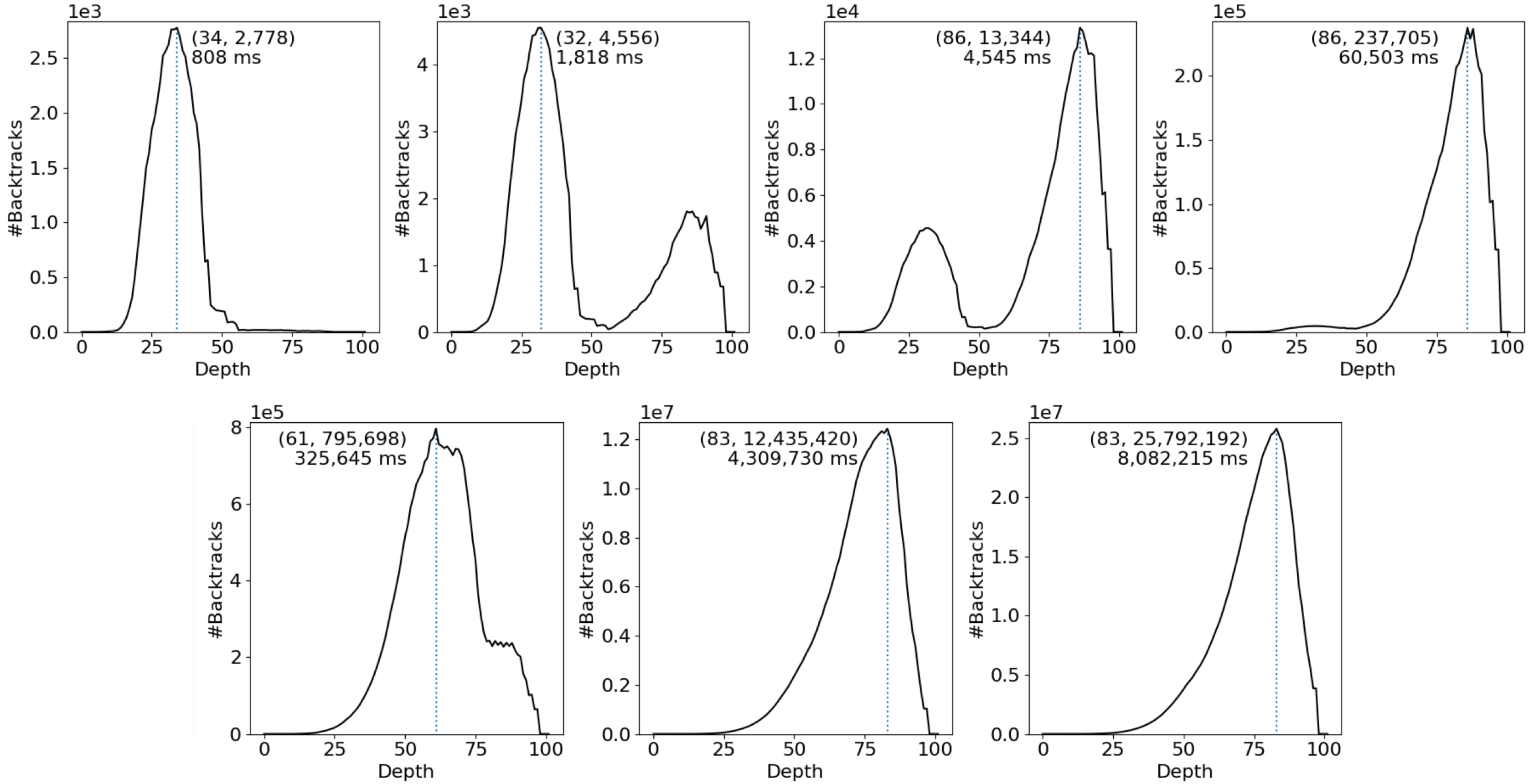
- Create a summarization tree
  - Cut the tree at a user-defined δ or
  - Cut the tree to include a n *regimes*



Example: Analyzing structure

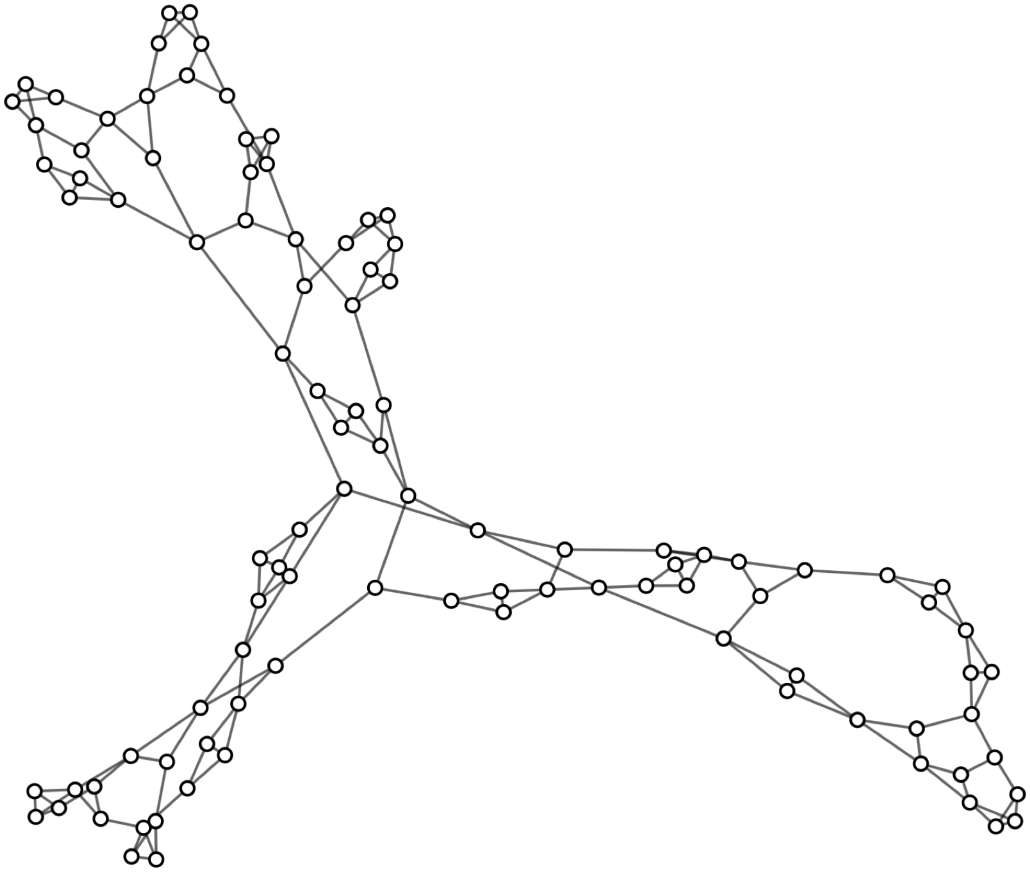


Example: Analyzing structure



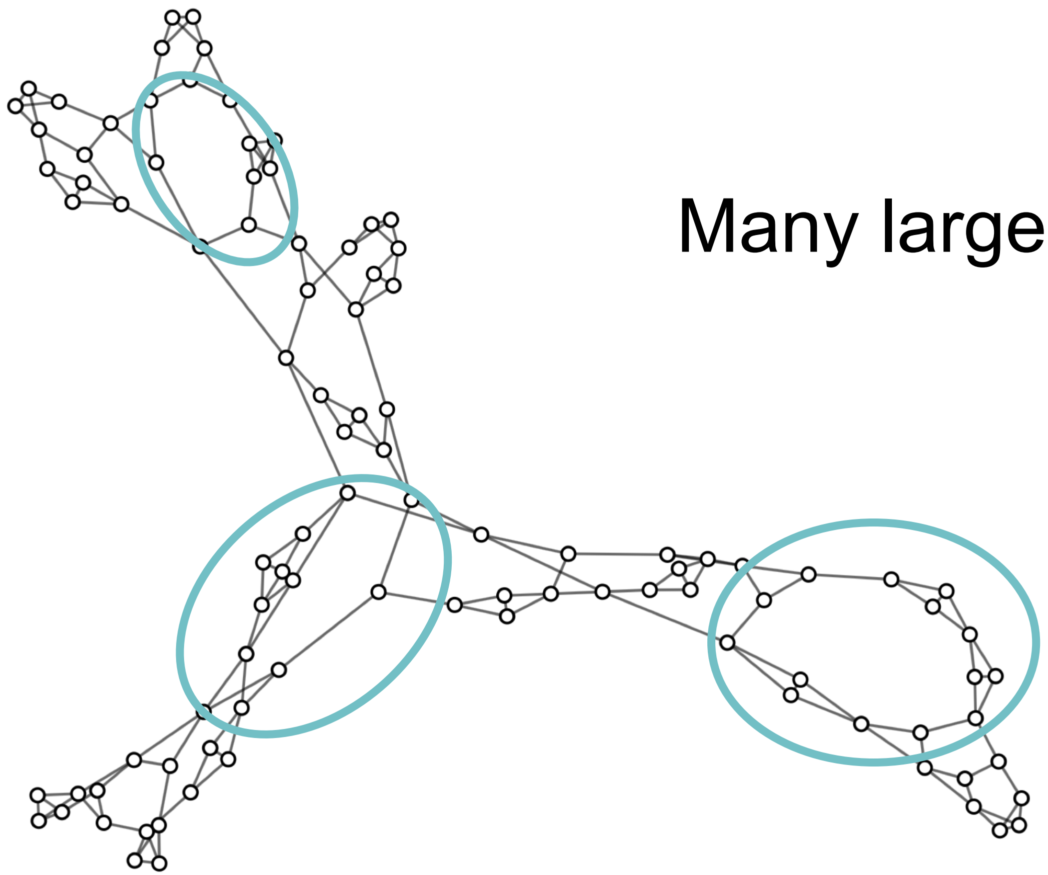
0.1-summarization of GAC on MUG100-25-3

Example: Analyzing structure



Primal graph of MUG100-25-3

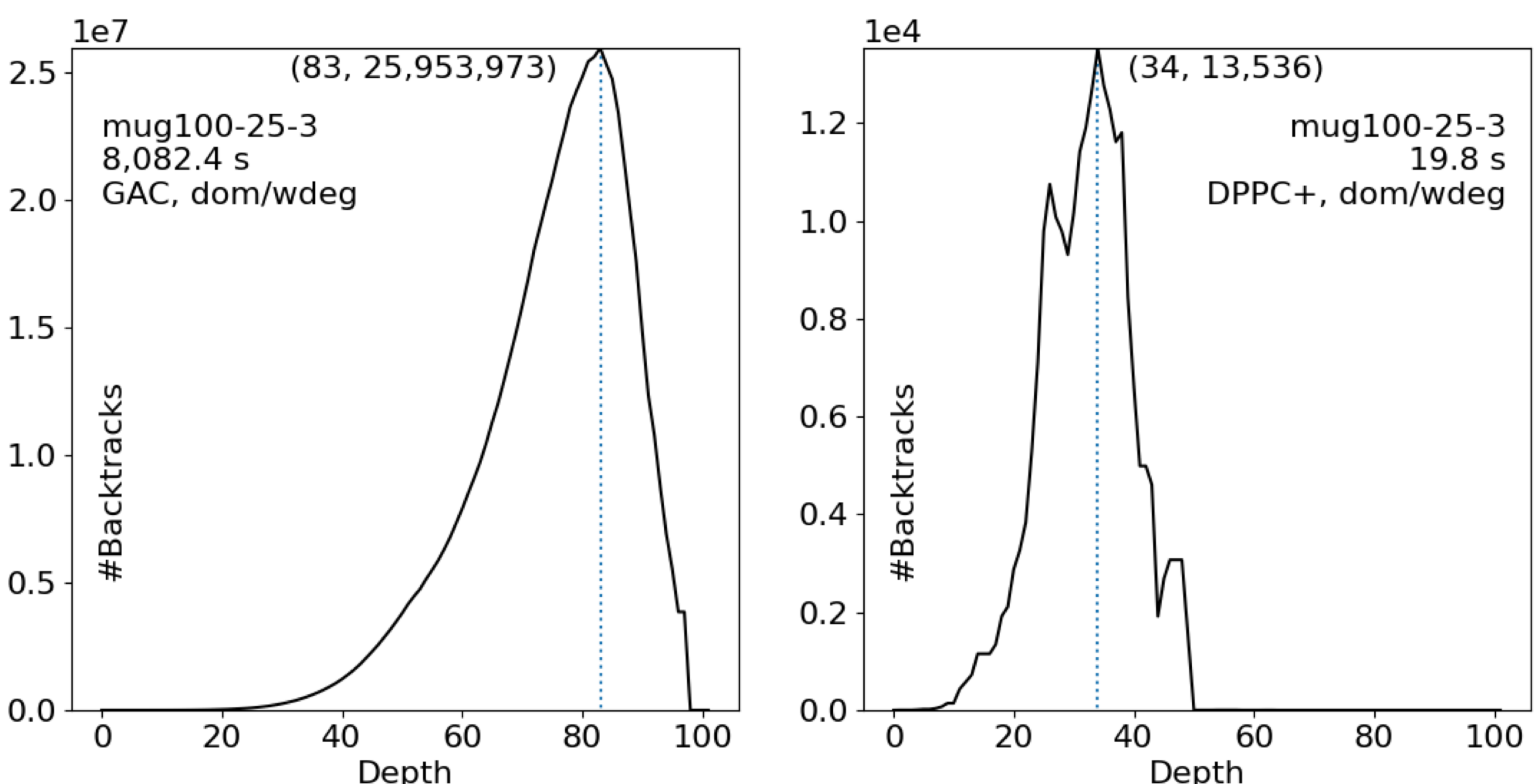
Example: Analyzing structure



Many large cycles

Primal graph of MUG100-25-3

Example: Analyzing structure



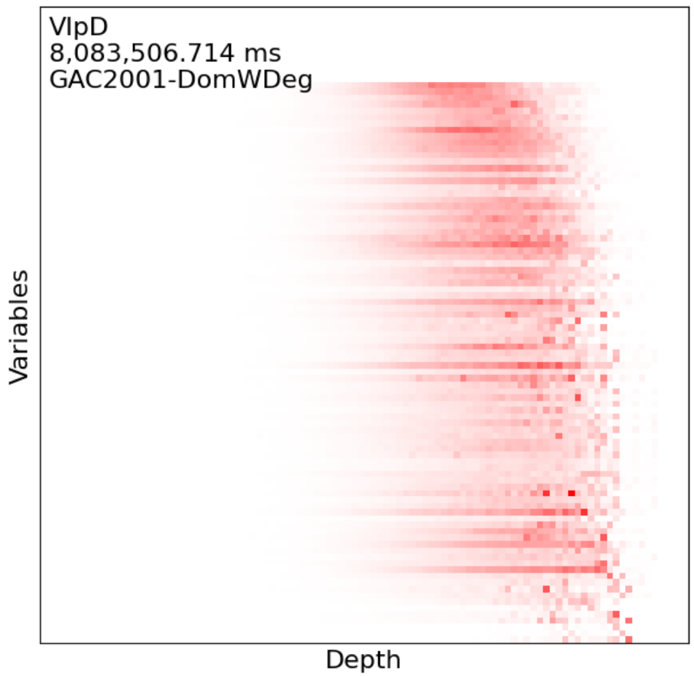
BpD of GAC and DPPC+ on MUG100-25-3

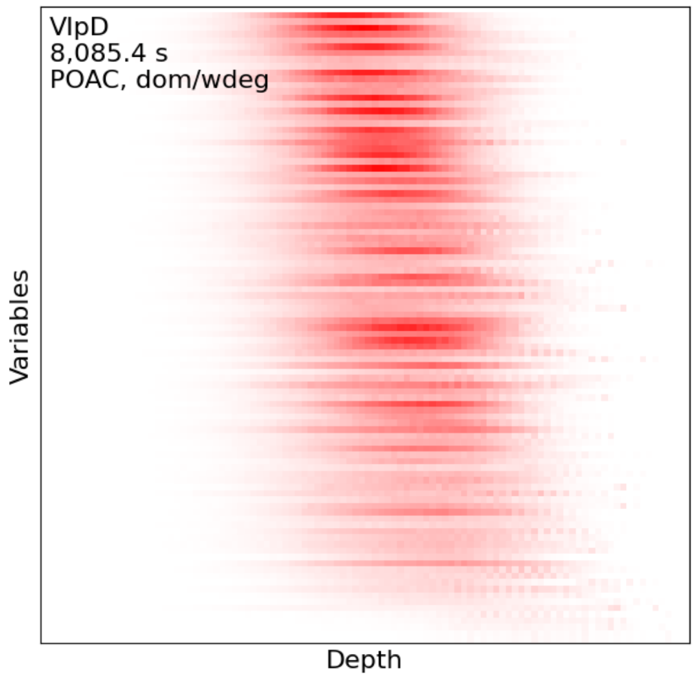
Visualizing variable ordering

- Variable Instantiations per Depth (VIpD)
  - $I(v,d,t)$ is the number of instantiations of variable v at depth d and time t
- Order the variables of the VIpD according to each variable's weighted depth:

$$d_w(v,t) = \frac{\sum_{d \in D} I(v,d,t) \cdot d}{\sum_{d \in D} I(v,d,t)}$$

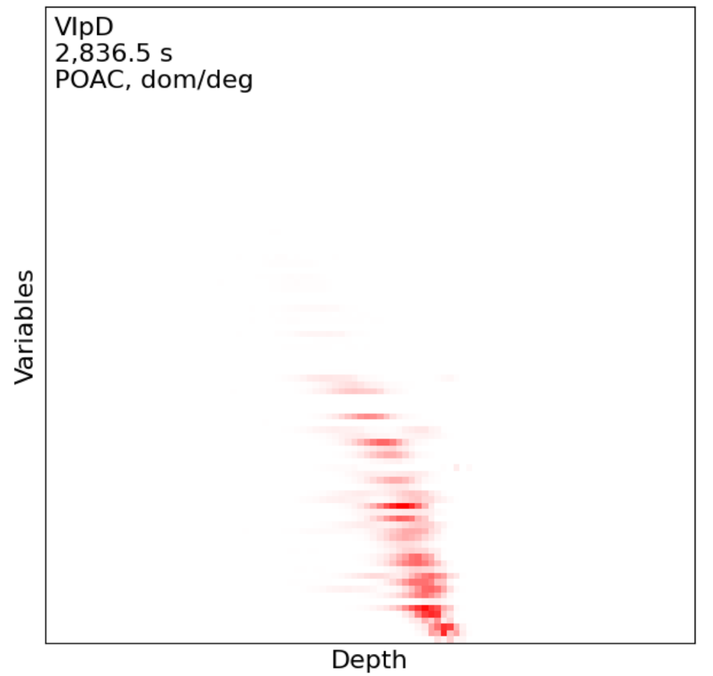
Example: Variable ordering





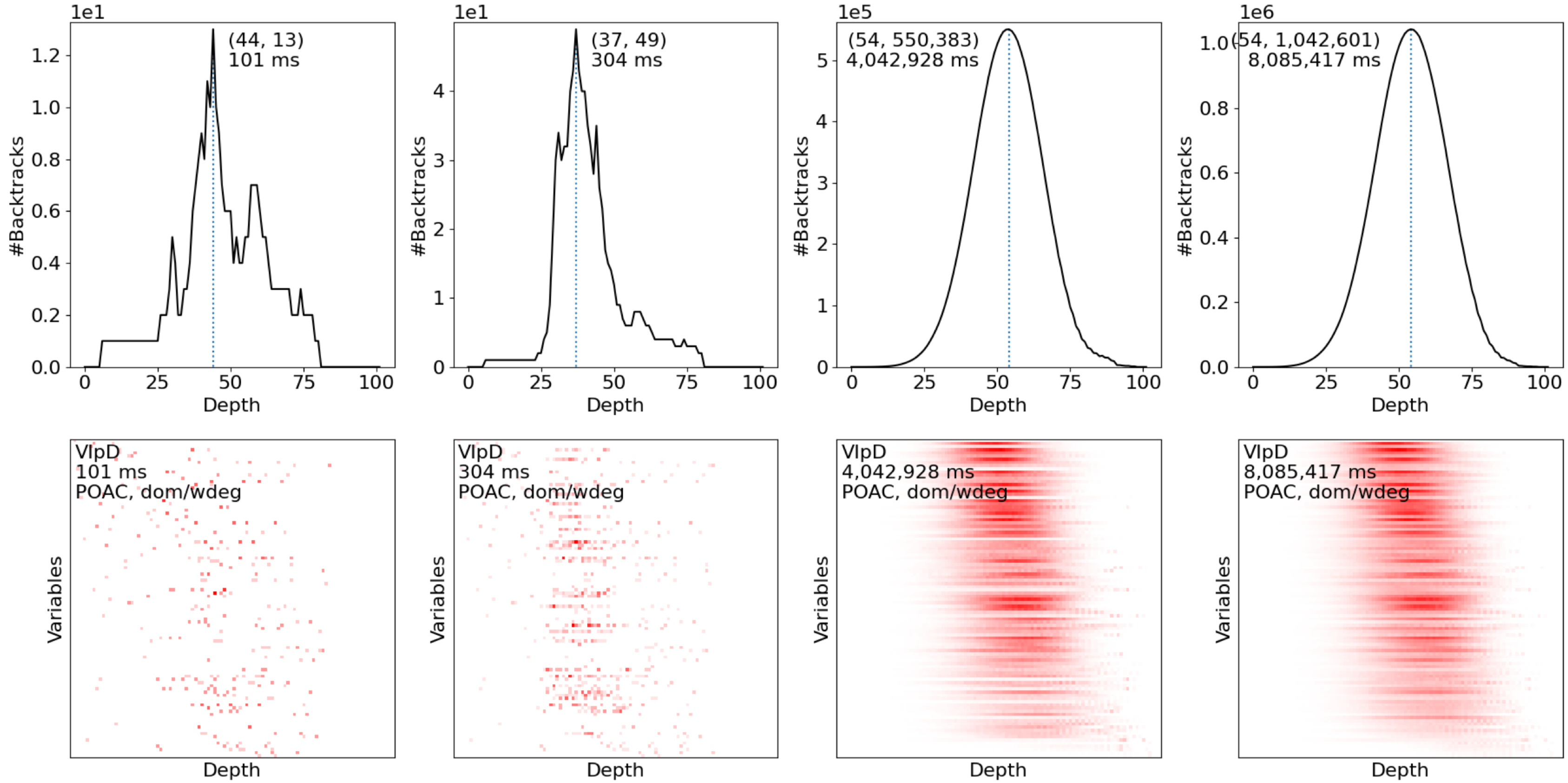
VIpD of GAC and POAC with dom/wdeg on MUG100-1-3

Example: Variable ordering



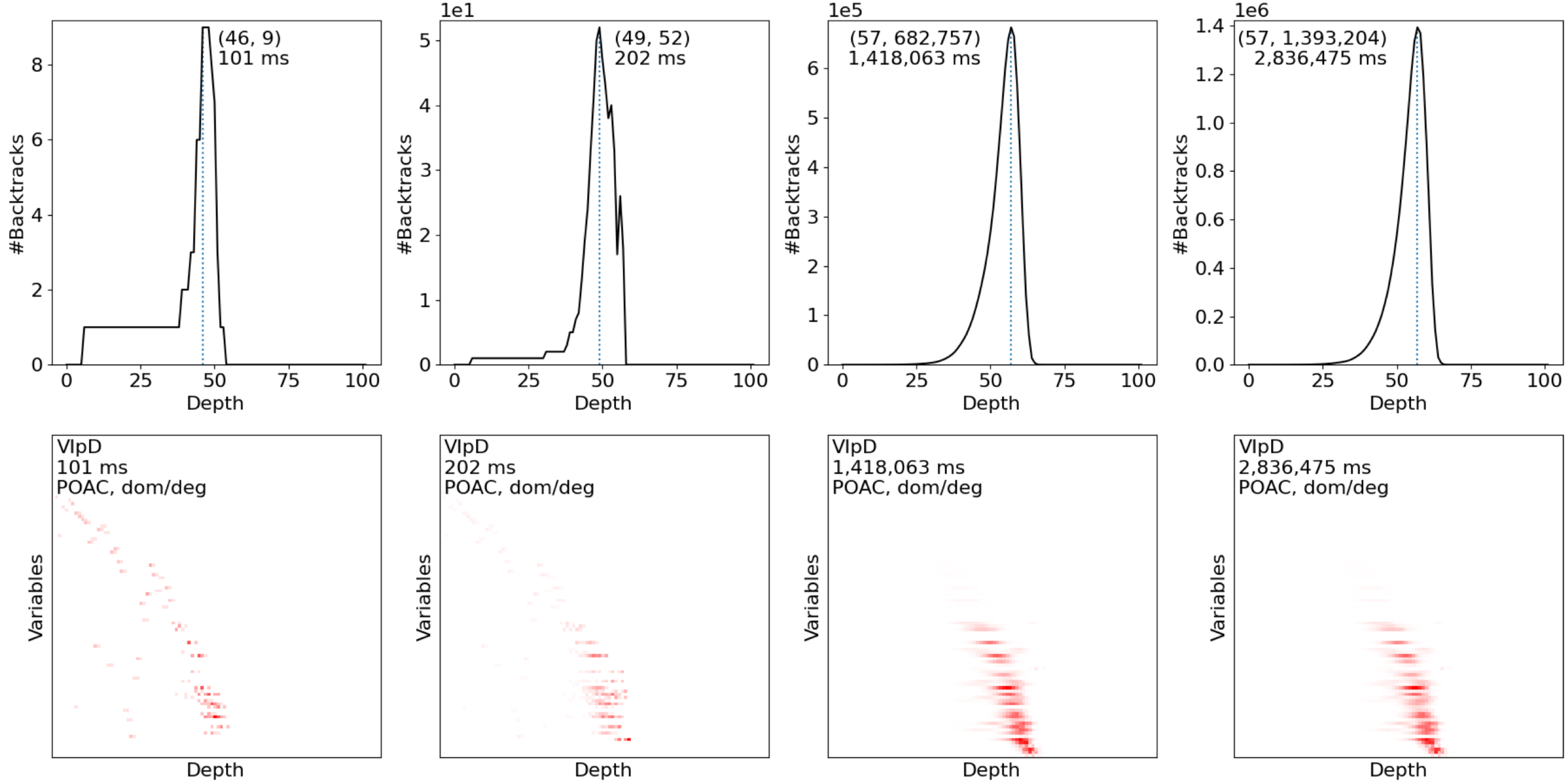
VlpD of POAC with dom/deg on MUG100-1-3

Example: Variable ordering



0.1-summarization of POAC with dom/wdeg on MUG100-1-3

Example: Variable ordering



0.1-summarization of POAC with dom/deg on MUG100-1-3

Conclusions and discussion

- Summarizations can be used to help explain the (changing) behavior of search
- Researchers and developers can use these tools iteratively to better study the impact of a strategy on a given problem
- Summarizations can catch small and large behavior that a human could not

Conclusions and discussion

- Initial results of BpD with binary branching give similar results
- We currently provide a ‘post-mortem’ analysis. Future work includes an ‘in-vivo’ analysis that enables human experts

Acknowledgements

- This research is supported by NSF Grant No. RI-1619344 and NSF CAREER Award No. III-1652846.
- The experiments were completed utilizing the Holland Computing Center of the University of Nebraska, which receives support from the Nebraska Research Initiative.