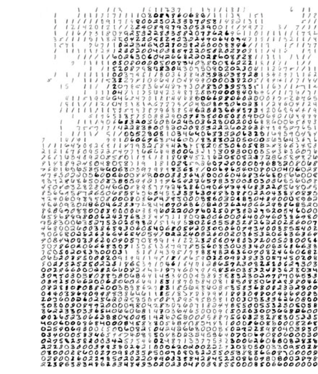
Human Priors in Hierarchical Program Induction



Mark K Ho*, Sophia Sanborn*, Frederick Callaway*, David Bourgin, and Thomas L. Griffiths

University of California, Berkeley

mark_ho@berkeley.edu, sanborn@berkely.edu, fredcallaway@berkeley.edu

Background

- Human problem-solving behavior is organized into **rich hierarchical structure** [1, 2].
- Inferring this structure is essential for interpreting behavior and anticipating how others will act.

Research Goals

- Previous work cast hierarchy learning as an efficient coding problem and found people often generate shorter programs to solve problems [3].
- Here, we examine alternative **program features** that constrain how people **interpret** hierarachically organized behavior.

Programs as Problem Solutions

- A program (π) is a set of subprocesses, σ_i, which are sequences of **primitive actions** or **calls to other subprocesses**.

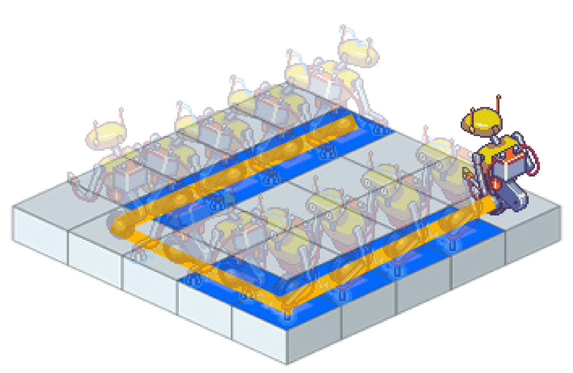
$$\pi = \{\sigma_{\text{Root}}, \sigma_1, \sigma_2\}$$

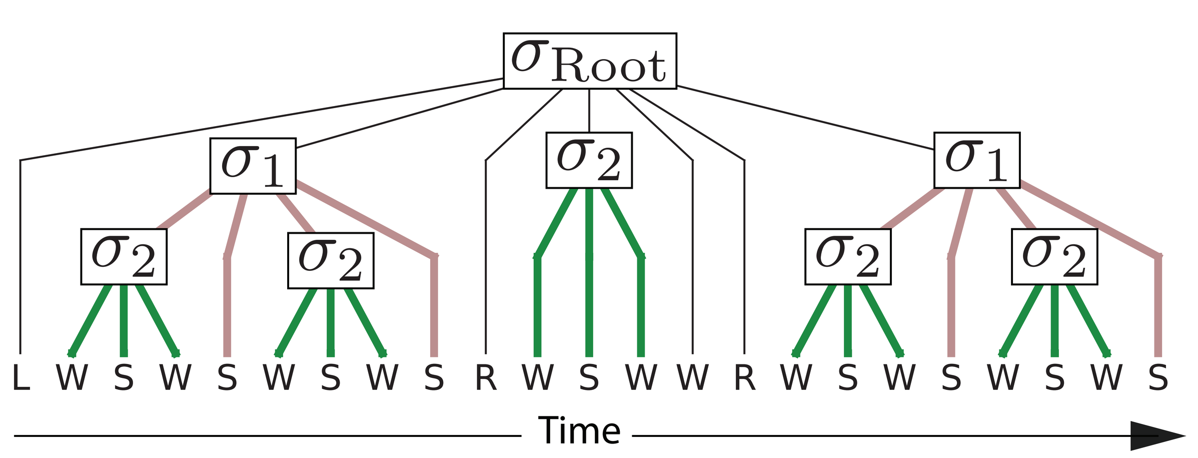
$$\sigma_{\text{Root}} = (\texttt{Left}, \sigma_1, \texttt{Right}, \sigma_2, \texttt{Walk}, \texttt{Right}, \sigma_1)$$

$$\sigma_1 = (\sigma_2, \texttt{Switch}, \sigma_2, \texttt{Switch})$$

$$\sigma_2 = (\texttt{Walk}, \texttt{Switch}, \texttt{Walk})$$

- Executing a program produces a **state-action trace** as well as an **execution tree**.





References

[1] Simon, H. A. (1991) The Architecture of Complexity.

[2] Solway, A., Diuk, C., Cordova, N., Yee, D., Barto, A., Niv, Y., & Botvinick, M. (in press). Optimalbehavioral hierarchy. PLOS Computational Biology.

[3] Sanborn, S., Bourgin, D., Chang, M., & Griffiths, T. (2018). Representational efficiency outweighs action efficiency in human program induction. In Proceedings of the 40th annual cognitive science society.

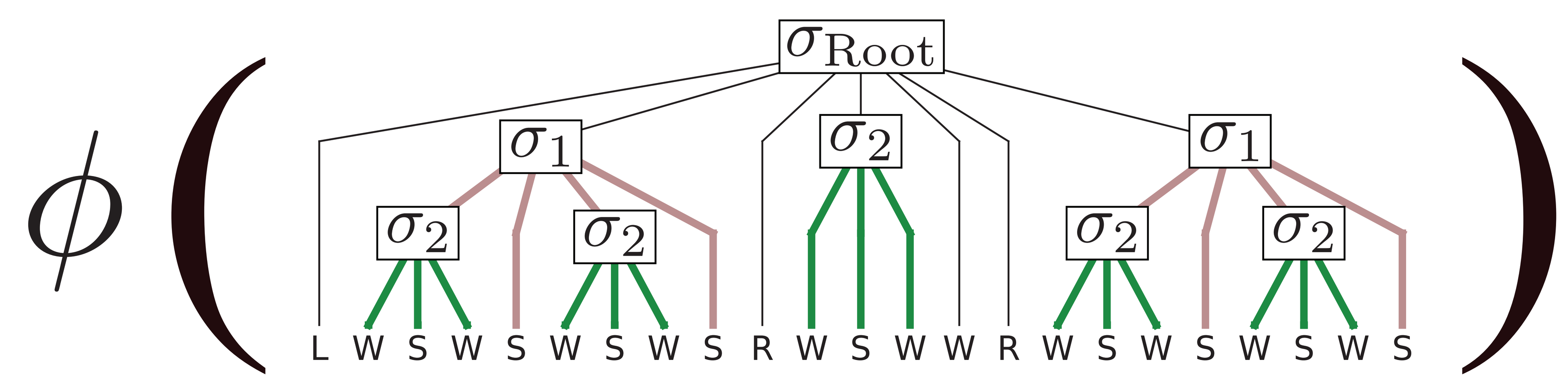
Inferring Programs

- Inducing a program from a trace can be expressed as probabilistic inference:

$$p(\pi \mid \zeta) = \frac{p(\zeta \mid \pi)p(\pi)}{\sum_{\pi'} p(\zeta \mid \pi')p(\pi')}$$

- The program prior is a function of weighted program features.

$$p(\pi; \theta) \propto \exp\{\theta^\top \phi(\pi)\}$$



$=$

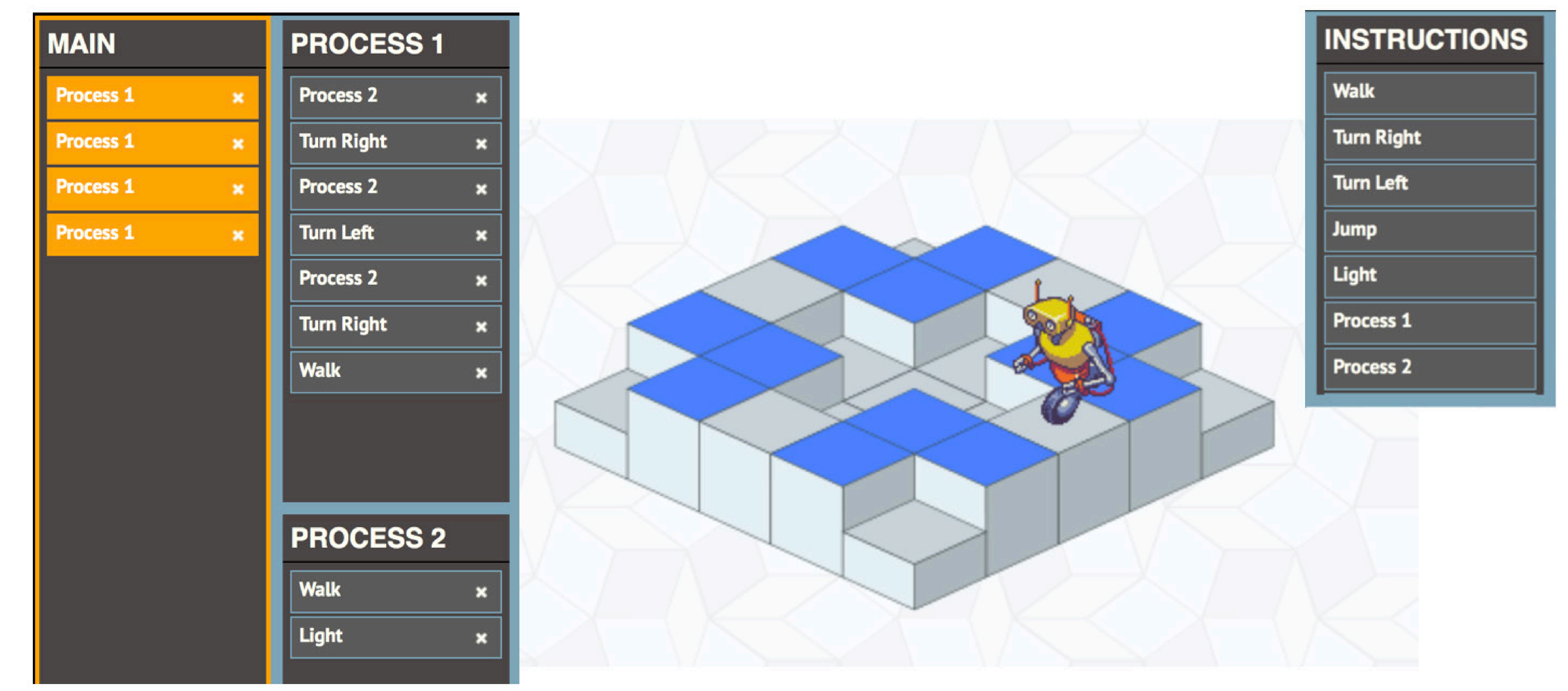
14	Total Program Length
3	Execution Tree Depth
2	Number of Subprocesses
1.7	Subprocess Length (S.D.)
0.66	Action/Subprocess Entropy (M)
0.81	Action-Call Entropy (M)
0.46	Process-Call Entropy (M)
7	Root Length
1.67	Children per Subprocess (M)
1.5	Parents per Subprocess (M)
1.38	Subprocess Entropy (M)
0.58	Subprocess to Action Ratio (M)

- Given features ϕ, trace ζ, and program π, we want to estimate the feature weights θ

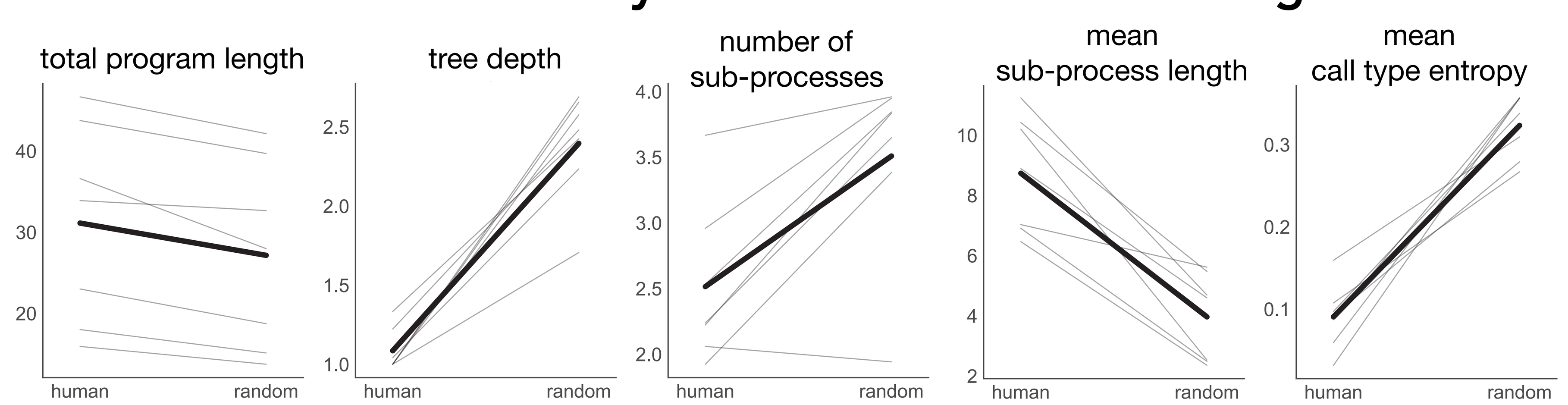
$$p(\theta \mid \pi, \zeta) \propto \frac{p(\zeta \mid \pi)p(\pi; \theta)}{p(\zeta; \theta)} p(\theta)$$

Estimating Human Priors

- 77 participants observed human-generated solution traces in the Lightbot domain [1] and tried to reconstruct the original programs that generated the traces.



Empirical Feature Weights for Humans vs. Randomly Generated Consistent Programs



Conclusions

- Participants prefer programs that have shallower trees and use fewer, longer subprocesses.
- These findings may reflect working memory constraints that limit the structural complexity of induced programs.
- In ongoing work, we are investigating the relationships between these features and their individual contributions to program induction, program generation, and perceived program complexity.