Perception-based Grammatical Inference for Adaptive Systems

Jie Fu (U. Pennsylvania)

Jeffrey Heinz (Delaware)

Adam Jardine (Delaware)

Herbert G. Tanner (Delaware)

The 12th International Conference of Grammatical Inference

University of Kyoto, Japan

September 19, 2014

The researchers from Delaware acknowledge support from NSF#1035577.

This paper

1. We introduce a learning paradigm called sensor-*identification in the limit from positive data*.
2. sensor is a perception module that obfuscates the learner's input.
3. Exact identification is eschewed for converging to a grammar which generates a language approximating the target language.
4. Successful approximation is understood as matching up to *observation-equivalence*.
5. Theoretical work exists which addresses other kinds of imperfect presentations, oracles, and the kinds of results obtainable with them [AL88, Ste95, FJ96, CJ01, THJ06].

Motivation (part I)

1. A frontier in robotics is managing uncertainty.
2. Earlier work showed how to use grammatical inference to reduce the uncertainty in environments with potentially adversarial, but rule-governed behavior [CFK$^+$12, FTH13, FTHC14].
3. The robot's capabilities, task, and environment were modeled as finite-state transition systems and product operations brought these elements together to form a *game*, allowing optimal control strategies to be computed (if they exist).
4. However, that work assumed *perfect* information about the environment.

Motivation (part II)

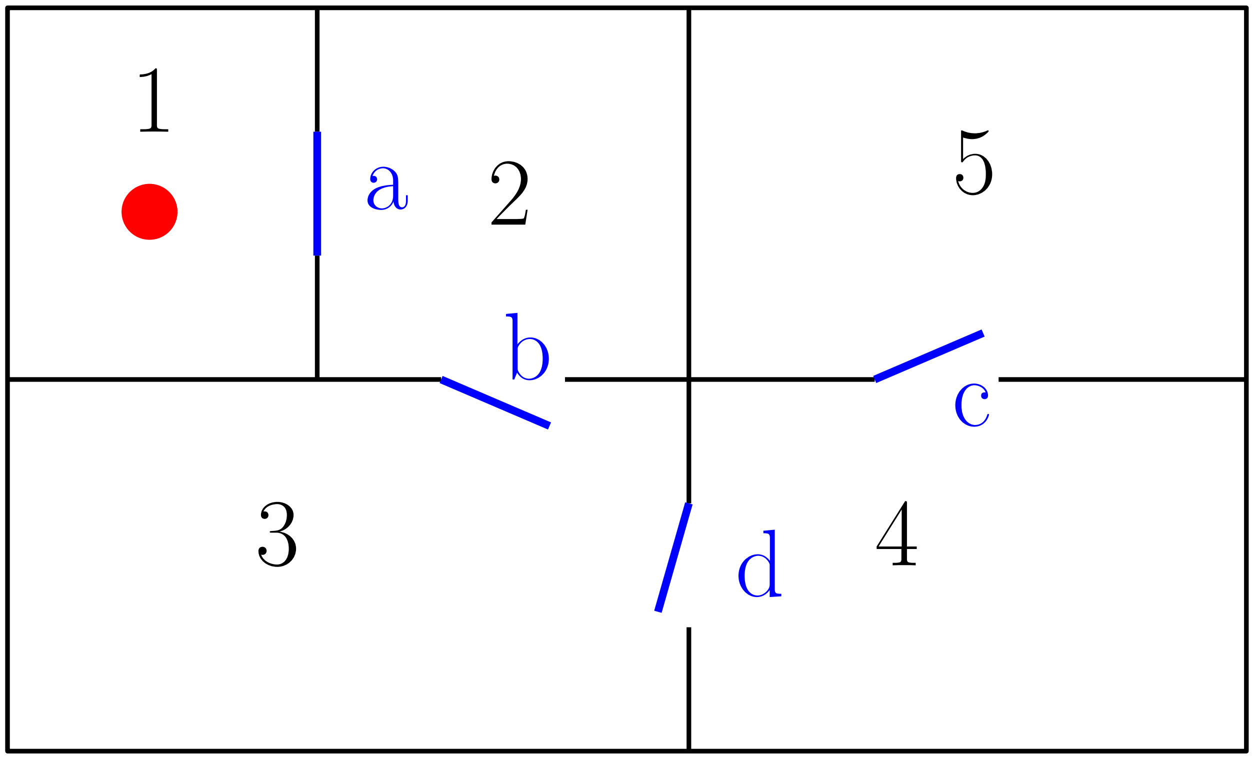
1. Recent results in game theory [AVW03, CDHR06] shows that optimal strategies can be found even for games with *imperfect* information (where players only have partial information about the state of the game).

2. The techniques in [CFK$^+$12, FTH13, FTHC14] allow *imperfect* games to be constructed from *imperfect*—but consistent—models of the environment.

3. What is missing then is a way to identify such models from *imperfect* observations.

4. (POMDPs and MDPs address 1-player stochastic games, not 2-player games.)

Motivating Example



Basic Strategy

1. Convert learning solutions in the identification in the limit from positive data paradigm to solutions in the sensor-identification paradigm.

2. We focus on learnable regular classes of languages, which are well-studied [dlH10].

Sensor models

Sensor models have been proposed [CL08, LEPDG11, FDT14]. The definition below subsumes them all.

A sensor model is $\mathsf{sensor} = \langle \Theta, \Sigma, \sim_\theta \ (\forall \theta \in \Theta), L_\Theta \rangle$ where

- Θ and Σ are finite, ordered sets of alphabets (the former being the *sensor configurations*).
- For all $\theta \in \Theta$, $\sim_\theta$ is an equivalence relation on Σ. If $\sigma_1 \sim_\theta \sigma_2$ then σ_1 is *indistinguishable* from σ_2 under sensor configuration θ. Let $[\sigma]_\theta = \{\sigma' \in \Sigma \mid \sigma' \sim_\theta \sigma\}$.
- $L_\Theta \subseteq \Theta^*$ is regular and represents the permissible sequences of sensor configurations.

We let $\hat{\Sigma}$ denote the powerset of Σ. So $[\sigma]_\theta \in \hat{\Sigma}$.

Observations (part I)

1. A bi-word is an element of $(\Theta \times \Sigma)^*$.

2. Let π_1 and π_2 be the left and right projections of $w \in (\Theta \times \Sigma)^*$.

3. $\mathsf{obs} : (\Theta \times \Sigma)^* \to \hat{\Sigma}^*$ is defined inductively as follows.

   - The base case: $\mathsf{obs}(\lambda) = \{\lambda\}$.
   - The inductive case:

$$\mathsf{obs}(w \cdot (\theta, \sigma)) = \mathsf{obs}(w) \cdot [\sigma]_\theta$$

4. Thus $\mathsf{obs}(u, v)$ is the *finite* set of sequences in Σ^* that are indistinguishable from v given the sequence u of sensor configurations.

Running Example (1)

Let $\Theta = \{\theta\}, \Sigma = \{0, 1, 2\}$, and $[0]_\theta = \{0\}$ and $[1]_\theta = [2]_\theta = \{1, 2\}$.

Consider the biword $w = (\theta, 0)(\theta, 1)(\theta, 1)(\theta, 0)(\theta, 2)(\theta, 2)$.

Then:

1. $\pi_1(w) = \theta\theta\theta\theta\theta\theta$.
2. $\pi_2(w) = 011022$.
3. $$\begin{aligned} \mathsf{obs}(w) &= [0]_\theta\ [1]_\theta\ [1]_\theta\ [0]_\theta\ [2]_\theta\ [2]_\theta \\ &= \{0\}\{1,2\}\{1,2\}\{0\}\{1,2\}\{1,2\} \end{aligned}$$

Observations (part II)

Similarly, each $u \in \Theta^*$, a sensor model inductively induces an equivalence relation $\sim_u$ over Σ^*.

- The base case: $\lambda \sim_\lambda \lambda$
- The inductive case: $(\forall \sigma_1, \sigma_2 \in \Sigma, v_1, v_2 \in \Sigma^*, \theta \in \Theta, u \in \Theta^*)$

$$\left[v_1 \sim_u v_2 \Rightarrow (v_1\sigma_1 \sim_{u\theta} v_2\sigma_2 \Leftrightarrow \sigma_1 \sim_\theta \sigma_2)\right]$$

Let $[v]_u = \{v' \in \Sigma^* \mid v \sim_u v'\}$, which denotes equivalent strings in Σ^* according to $u \in \Theta^*$.

Lemma 1. *For all $w \in (\Theta \times \Sigma)^*$, $[\pi_2(w)]_{\pi_1(w)} = \mathsf{obs}(w)$ is a finite subset of Σ^*.*

Running Example (2)

Consider biwords

$$w_1 = (\theta, 0)(\theta, 1)(\theta, 1)(\theta, 0)(\theta, 2)(\theta, 2)$$

$$w_2 = (\theta, 0)(\theta, 2)(\theta, 1)(\theta, 0)(\theta, 1)(\theta, 2)$$

Then

1. $\mathsf{obs}(w_1) = \mathsf{obs}(w_2)$
2. $w_1 \sim_{\theta\theta\theta\theta\theta\theta} w_2$

Facts and Observations

Facts on the Ground

Given L_Θ and L_Σ, the *facts on the ground* are

$$L_{\mathsf{system}} \stackrel{\text{def}}{=} \left\{ w \in (\Theta \times \Sigma)^* \mid \pi_1(w) \in L_\Theta \text{ and } \pi_2(w) \in L_\Sigma \right\}$$

The Observations on the Ground

In contrast, the *observations on the ground* are:

$$\begin{aligned} L_{\mathsf{sensor}} \quad \stackrel{\text{def}}{=} \quad \Big\{ \quad & \hat{w} \in (\Theta \times \hat{\Sigma})^* \mid \exists w \in L_{\mathsf{system}} \text{ and} \\ & \pi_1(\hat{w}) = \pi_1(w) \text{ and } \pi_2(\hat{w}) = \mathsf{obs}(w) \quad \Big\} \end{aligned}$$

Running Example (3)

Consider the languages

$$
\begin{aligned}
L_\Theta &= \theta^* \\
L_\Sigma &= \Big\{ w \;\Big|\; |w|_0, |w|_1, |w|_2 \text{ are each even} \Big\}
\end{aligned}
$$

Then

1. $w_1 = (\theta, 0)(\theta, 1)(\theta, 1)(\theta, 0)(\theta, 2)(\theta, 2)$ and
   $w_2 = (\theta, 0)(\theta, 2)(\theta, 1)(\theta, 0)(\theta, 1)(\theta, 2)$ belong to L_{system}.
2. $\Big(\theta, \{0\}\Big)\Big(\theta, \{1,2\}\Big)\Big(\theta, \{1,2\}\Big)\Big(\theta, \{0\}\Big)\Big(\theta, \{1,2\}\Big)\Big(\theta, \{1,2\}\Big)$ is an element of L_{sensor}.

Observation-equivalence of Languages

Definition 1 (Observation-equivalence)**.** *According to model* sensor, *languages* $L, L' \subseteq \Sigma^*$ *are* observation-equivalent *if*

$$(\forall v \in L)(\exists v' \in L')(\forall u \in \{u \mid (u, v) \in L_{\mathsf{system}}\})\big[v \sim_u v'\big] \qquad \textit{and}$$

$$(\forall v' \in L')(\exists v \in L)(\forall u \in \{u \mid (u, v') \in L'_{\mathsf{system}}\})\big[v \sim_u v'\big]$$

Running Example (4)

Fix $L_\Theta = \theta^*$. Consider

$$\begin{array}{rcl} L_t & = & \{w \mid |w|_0, |w|_1, |w|_2 \text{ are each even}\} \\ L_h & = & \{w \mid |w|_0, \big(|w|_1 + |w|_2\big) \text{ are both even}\} \end{array}$$

Then

1. L_t is observation-equivalent to L_h.

Illustration: Let $w_3 = (\theta,1)(\theta,1)(\theta,1)(\theta,2)(\theta,2)(\theta,2)$.

Then $\pi_2(w_3) = 111222 \in L_h$ but $\pi_2(w_3) \notin L_t$. Nonetheless, $\mathsf{obs}(w_3) = \{1,2\}\{1,2\}\{1,2\}\{1,2\}\{1,2\}\{1,2\}$ and there exists w_4 such that $\pi_2(w_4) = 112211 \in L_t$ such that $\mathsf{obs}(w_4) = \mathsf{obs}(w_3)$.

Sensor-identification in the limit

We consider a sensor model $\mathsf{sensor} = \langle \Theta, \Sigma, \sim_\theta \ (\forall \theta \in \Theta), L_\Theta \rangle$ and family of languages $\mathcal{L}$ over Σ.

$\mathcal{L}$ is sensor-*identifiable in the limit from positive data* if there exists an algorithm $\mathfrak{A}$ such that for all $L \in \mathcal{L}$, for any presentation ϕ of L_{sensor}, there exists $n \in \mathbb{N}$ such that for all $m \geq n$,

- $\mathfrak{A}(\phi[m]) = \mathfrak{A}(\phi[n]) = G$, and **(convergence)**
- $L(G)$ is observation-equivalent to L. **("correctness")**

Running Example (5)

If the target language is this one:

$$L_t = \left\{ w \;\middle|\; |w|_0, |w|_1, |w|_2 \text{ are each even} \right\}$$

Then presentations draw elements from L_{sensor}:

$$\text{not } (\theta,0)(\theta,0)(\theta,1)(\theta,1)(\theta,2)(\theta,2) \text{ but}$$

$$\Big(\theta,\{0\}\Big)\Big(\theta,\{0\}\Big)\Big(\theta,\{1,2\}\Big)\Big(\theta,\{1,2\}\Big)\Big(\theta,\{1,2\}\Big)\Big(\theta,\{1,2\}\Big)$$

$$\text{not } (\theta,1)(\theta,0)(\theta,2)(\theta,0)(\theta,1)(\theta,2) \text{ but}$$

$$\Big(\theta,\{1,2\}\Big)\Big(\theta,\{0\}\Big)\Big(\theta,\{1,2\}\Big)\Big(\theta,\{0\}\Big)\Big(\theta,\{1,2\}\Big)\Big(\theta,\{1,2\}\Big)$$

$$\ldots$$

Learning regular languages

For any L, let $\sim_L$ be the Myhill-Nerode equivalence relation for L.

$$w \sim_L w' \Leftrightarrow \{v \in \Sigma^* \mid wv \in L\} = \{v \in \Sigma^* \mid w'v \in L\}$$

.

1. Given as input a finite sample $S \subset \Sigma^*$, a learning algorithm $\mathfrak{A}$ determines an equivalence relation $\sim_{\mathfrak{A}}$ over Σ^*.
2. For any regular L, for any presentation ϕ of L, if $\mathfrak{A}(\phi)$ outputs $\sim_{\mathfrak{A}}$, which is of finite index and refines $\sim_L$ then $\mathfrak{A}$ identifies L in the limit from positive data.
3. If $\mathfrak{A}$ does this for every $L \in \mathcal{L}$ then $\mathfrak{A}$ identifies $\mathcal{L}$ in the limit from positive data.

Useful Lemma

Lemma 2. *If L_Θ and L are regular then $\sim_{L_{\mathsf{system}}}$ is of finite index and a right congruence. Furthermore,*

$$w \sim_{\mathsf{system}} w' \Leftrightarrow \pi_1(w) \sim_{L_\Theta} \pi_1(w') \wedge \pi_2(w) \sim_L \pi_2(w')$$

Lifting congruences to $\hat{\Sigma}^*$

A right congruence $\sim$ over Σ^* induces a relation $\approx$ among elements of $\mathscr{P}(\Sigma^*)$:

$$X \approx Y \Leftrightarrow (\forall x \in X)(\exists y \in Y)(x \sim y) \wedge (\forall y \in Y)(\exists x \in X)[x \sim y]$$

Since elements of $\hat{\Sigma}^*$ can be understood as subsets of Σ^*, $\approx_L$ is meaningful on $\hat{\Sigma}^*$.

Lemma 3. *If $\sim_{\mathsf{system}}$ is of finite index and a right congruence then so is $\sim_{\mathsf{sensor}}$. Furthermore,*

$$w \sim_{\mathsf{sensor}} w' \Leftrightarrow \pi_1(w) \sim_{L_\Theta} \pi_1(w') \wedge \pi_2(w) \approx_L \pi_2(w')$$

1. By Lemmas 2 and 3, there is a DFA A accepting L_{sensor}. A defines a class of languages $\mathcal{L}_{\mathsf{sensor}}$ over Σ.

2. Each $L \in \mathcal{L}_{\mathsf{sensor}}$ is obtained by replacing each label (which is an element of $\Theta \times \hat{\Sigma}$) of each transition in A with one element drawn from the label's right projection (thus the drawn element belongs to Σ).

3. These choices can be made consistently since Σ is ordered.

Lemma 4. *Any $L' \in \mathcal{L}_{\mathsf{sensor}}$ is observation-equivalent to L.*

Main result

Theorem 1. *Let $\mathcal{L}$ be identifiable in the limit from positive data by a state-merging algorithm $\mathfrak{A}$ and consider* $\mathsf{sensor} = \langle \Theta, \Sigma, \sim_\theta (\forall \theta \in \Theta), L_\Theta \rangle$. *There exists an algorithm $\mathfrak{B}$ which* Sensor*-identifies $\mathcal{L}$ in the limit from positive data.*

Proof Sketch Algorithm $\mathfrak{B}$ which takes as input a finite set $S \subset L_{\mathsf{sensor}}$ is defined from $\mathfrak{A}$ which identifies $\mathcal{L}$, the equivalence relations $\theta \in \Theta$ on Σ, and L_Θ.

$\mathfrak{B}$ builds a PTA for S and merges prefixes according to $\sim_\mathfrak{B}$, defined as follows:

$$\hat{w} \sim_\mathfrak{B} \hat{w}' \Leftrightarrow \pi_1(\hat{w}) \sim_{L_\Theta} \pi_1(\hat{w}') \wedge \pi_2(\hat{w}) \approx_\mathfrak{A} \pi_2(\hat{w}').$$

(continued. . .)

Proof sketch (con't)

$$\hat{w} \sim_{\mathfrak{B}} \hat{w}' \Leftrightarrow \pi_1(\hat{w}) \sim_{L_\Theta} \pi_1(\hat{w}') \wedge \pi_2(\hat{w}) \approx_{\mathfrak{A}} \pi_2(\hat{w}').$$

1. Since L_Θ is regular, we assume it is given in terms of its minimal DFA and so $\sim_{L_\Theta}$ can be computed.
2. Also, $\approx_{\mathfrak{A}}$ can be computed since $\sim_{\mathfrak{A}}$ can be computed and every $\mathsf{obs}(w)$ $(w \in L_{\mathsf{system}})$ is a finite set.
3. In the limit, $\sim_{\mathfrak{B}}$ is of finite index because $\sim_{\mathfrak{A}}$ is of finite index.
4. Also in the limit, $\sim_{\mathfrak{B}}$ refines $\sim_{\mathsf{sensor}}$ because $\sim_{\mathfrak{A}}$ refines $\sim_L$ in the limit and by definition of $\approx$.
5. Thus this acceptor recognizes the same language as L_{sensor}, and by Lemma 4, a language L' observation-equivalent to L can be obtained.
6. Convergence to L' is guaranteed by drawing least elements to find it.

Demonstration #1: Zero-reversible languages $\mathcal{L}_{ZR}$

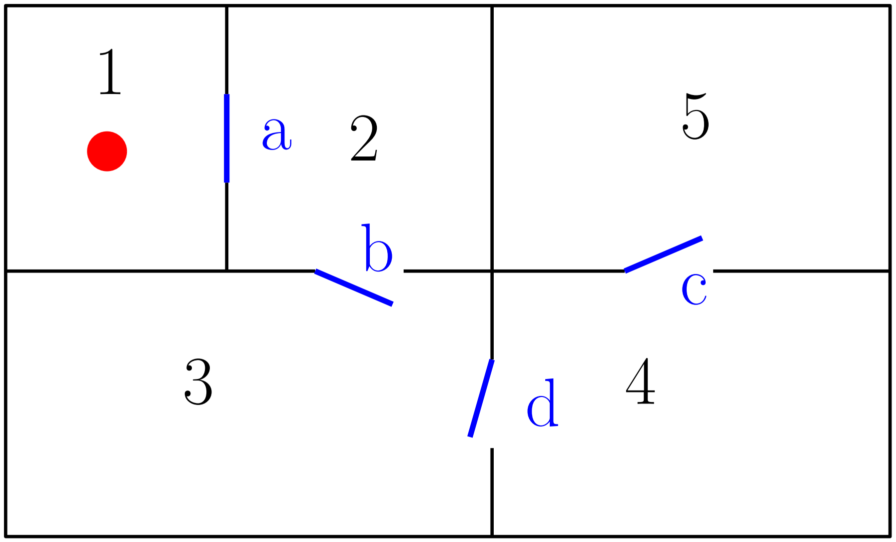
$$L_t = \{w \mid |w|_0, |w|_1, |w|_2 \text{ are each even}\} \in \mathcal{L}_{ZR}$$

With a sufficient sample, $\mathfrak{B}$ outputs a DFA recognizing this language.

$$L_h = \{w \mid |w|_0 \text{ and } (|w|_1 + |w|_2) \text{ are both even}\}$$

As mentioned, this hypothesized language L_h is observation-equivalent to the target L_t.

Demonstration #2: Robot motion planning



1. The game is turn based. The robot can only move to an adjacent room if the adjoining door is open.
2. The dynamic, adversarial environment opens and closes doors according to a Strictly 2-Local language. For instance perhaps the same door cannot be closed on consecutive terms.
3. The robot can only see which doors are open/closed which adjoin to the room it is in.

Conclusion

1. Using the aforementioned strategy, an observation-equivalent language can be learned.
2. Techniques described in [CFK$^+$12, FTH13, FTHC14] allow an *imperfect* game to be constructed.
3. Techniques from algorithmic game theory [AVW03, CDHR06] allow optimal strategies to be found.
4. Consequently, robots can deal with uncertainty better than before.

Thank you.

References

[AL88] Dana Angluin and Philip Laird. Learning from noisy examples. *Machine Learning*, 2:343–370, 1988.

[AVW03] André Arnold, Aymeric Vincent, and Igor Walukiewicz. Games for synthesis of controllers with partial observation. *Theoretical computer science*, 303(1):7–34, 2003.

[CDHR06] Krishnendu Chatterjee, Laurent Doyen, Thomas A Henzinger, and Jean-François Raskin. Algorithms for omega-regular games with imperfect information. In *Computer Science Logic*, pages 287–302. Springer, 2006.

[CFK+12] Jane Chandlee, Jie Fu, Konstantinos Karydis, Cesar Koirala, Jeffrey Heinz, and Herbert G. Tanner. Integrating grammatical inference into robotic planning. In Jeffrey Heinz, Colin de la Higuera, and Tim Oates, editors, *Proceedings of the Eleventh International Conference on Grammatical Inference*, volume 21, pages 69–83. JMLR Workshop and Conference Proceedings, August 2012.

[CJ01] J. Case and S. Jain. Synthesizing learners tolerating computable noisy data. *Journal of Computer and System Sciences*, 62:413–441, 2001.

[CL08] C. G. Cassandras and S. Lafortune. *Introduction to Discrete Event Systems*, volume 11. Springer, 2008.

[dlH10] Colin de la Higuera. *Grammatical Inference: Learning Automata and Grammars*. Cambridge University Press, 2010.

[FDT14] Jie Fu, Rayna Dimitrova, and Ufuk Topcu. Abstractions and sensor design in partial-information, reactive controller synthesis. In *American Control Conference, Porland, OR*, 2014.

[FJ96] M. Fulk and S. Jain. Learning in presence of inaccurate information. *Theoretical Computer Science*, 161:235–261, 1996.

[FTH13] Jie Fu, Herbert G. Tanner, and Jeffrey Heinz. Adaptive planning in unknown environments using grammatical inference. In *IEEE Conference on Decision and Control*, December 2013. To appear.

[FTHC14] Jie Fu, Bert Tanner, Jeffrey Heinz, and Jane Chandlee. Adaptive symbolic control for finite-state transition systems with grammatical inference. *IEEE Transactions on Automatic Control*, 59(2):505–511, February 2014.

[LEPDG11] Cai Luo, A.R. Espinosa, D. Pranantha, and A. De Gloria. Multi-robot search and