Efficient Control Synthesis and Learning in Distributed Cyber-Physical Systems

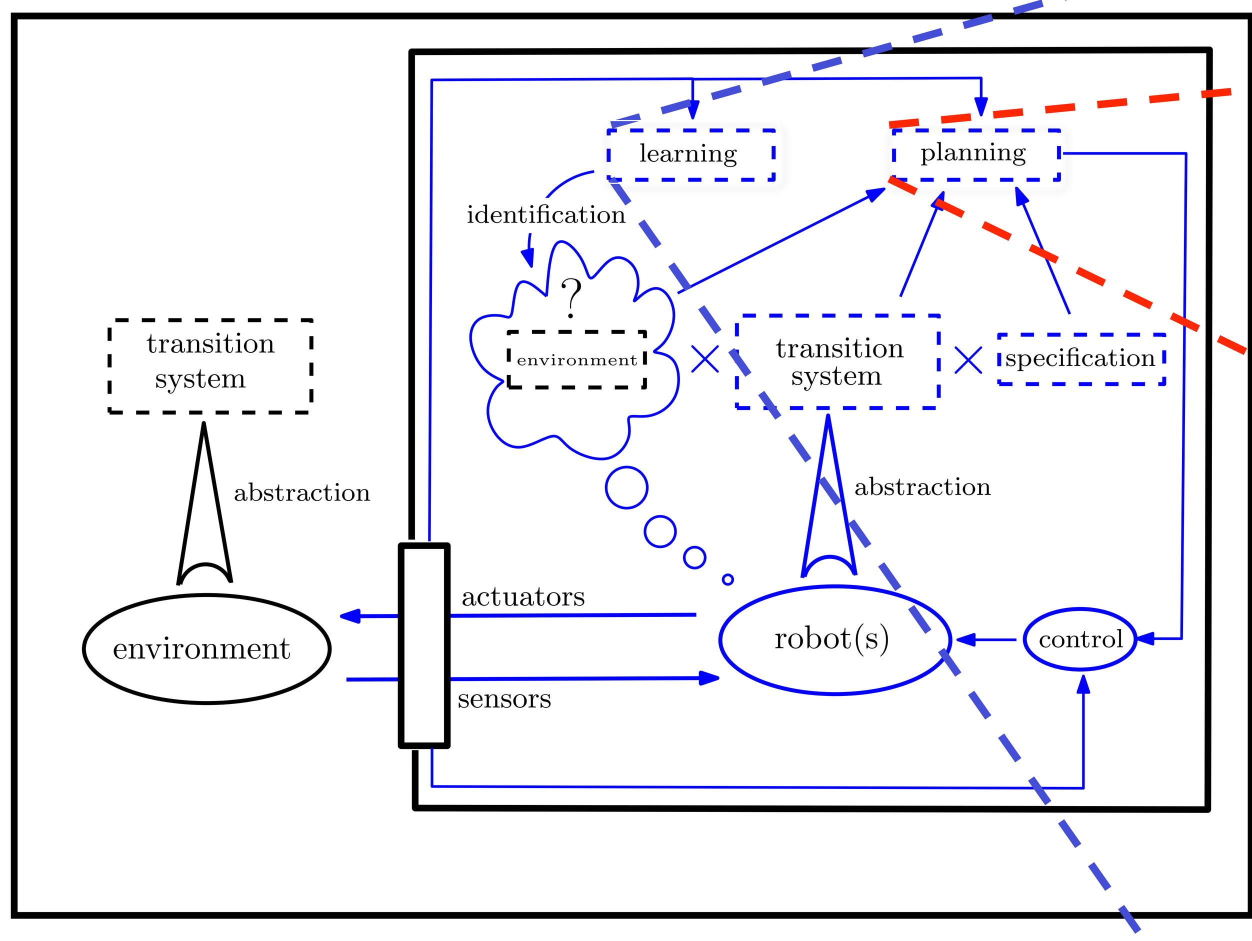
Herbert Tanner, Jeffrey Heinz
University of Delaware

Calin Belta
Boston University

Scientific Goal

Coordinate a group of heterogeneous autonomous cyber-physical systems to satisfy temporal logic control specifications in a partially unknown and dynamically changing environment.

A Bird's-Eye View



Learning and Adaptation

Goal

Learn from observations the behavior of the environment a multi-agent system interacts with while attempting to satisfy its specification.

Assumptions

- Knowledge of the class of formal languages the environment behavior falls into
- Fully observable environment evolution

Challenges

- Requirements for guaranteed asymptotic convergence of learning algorithm
- Full generalization without overfitting

Approach

- Formulate the problem as learning a repeated two-player turn-based game
- Adapt grammatical inference algorithms for learning games

Example

Suppose the dynamics of the unknown language can be modeled with a Strictly k-Piecewise (SP_k) language [7]. This class of languages is learnable with a string extension learner [8].

Strictly Piecewise Dynamics

- String $v = a_1 a_2 \ldots a_n$ is a subsequence of w iff $w \in \Sigma^* a_1 \Sigma^* a_2 \Sigma^* \ldots \Sigma^* a_n \Sigma^*$.
- Let $f_k(w) = \{v \mid v \text{ is a subsequence of } w \text{ and } |v| \leq k\}$.
- Example: $f_2(abacd) = \{\lambda, a, b, c, d, ab, aa, ac, ad, ba, bc, bd, cd\}$.
- $L \in \mathrm{SP}_k$ iff there exists a finite set $S \subseteq \Sigma^{\leq k}$ such that $f_k(L) = S$.
- This finite set can be viewed as the *grammar* generating L.

String Extension Learning

- A text T for L is an infinite sequence of elements of L such that each element of L occurs at least once in T.
- $T(i)$ is the ith element of T, and $T[i]$ is the finite sequence $T(1), T(2), \ldots T(i)$.
- Given any text T for any SP_k language L, the learning function ϕ_k converges to a grammar for L.

$$\phi_k(T[i]) = \begin{cases} \varnothing & i = 0 \\ \phi_k(T[i-1]) \cup f_k(T(i)) & \text{otherwise} \end{cases}$$

Temporal Logic Synthesis

Goal:
Synthesize a control policy to satisfy a mission specified using temporal logic

Example Mission:
"*Visit regions π_1 and π_2 infinitely often and always avoid the adversary*" (see Example of Scenario below)

Challenges:

- Ensure progress towards goal despite unknown actions by the adversary
- Computational issues arising from planning for multiple agents and adversary

Technical Approach:

- Translate mission specification from Linear Temporal Logic to a Büchi Automaton
- Construct Product Automaton from Game Transition System (see Learning and Adaptation at left) and Büchi Automaton to capture agent/environment interactions and satisfaction of specification
- Use "energy" function to determine distance to accepting states in the Product Automaton
  - Function is computed using backward induction
  - Adversary is assumed to choose most antagonistic actions
- Incrementally update the Product when new elements in the grammar are learned
  - Add new transitions in Game Transition System
  - For each new transition, add appropriate transitions in Product Automaton
  - Re-compute energy function
- At each step, control policy is the action that leads to a state with lower value for energy function, if one exists, otherwise report failure

Result:

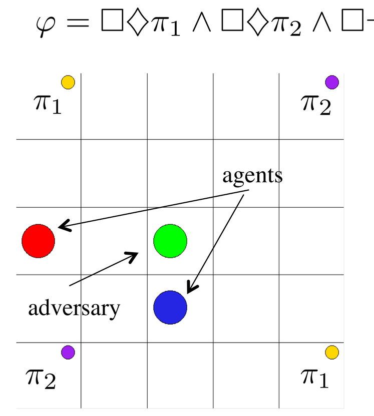
- Given that the learned model for adversary behavior is correct, our algorithm guarantees a control policy to satisfy the specification, if one exists

Integration of Learning and Control in Cyber Physical Systems Operating Under Uncertainty

Example of scenario

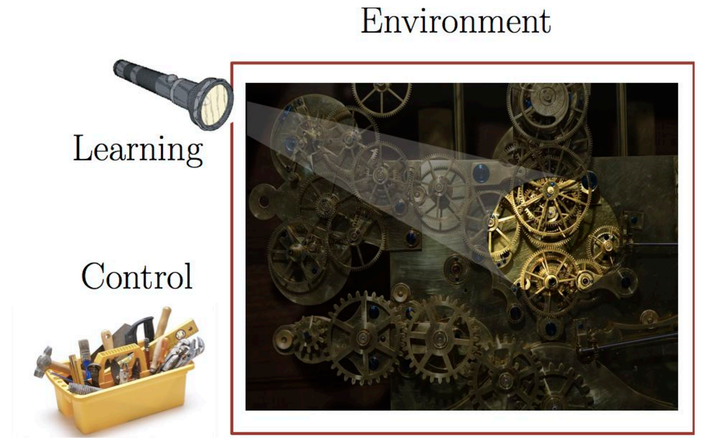
"Agents should visit the corners of the grid infinitely often while avoiding the adversary"

LTL : $\varphi = \Box\Diamond\pi_1 \wedge \Box\Diamond\pi_2 \wedge \Box\neg\pi_{\text{capture}}$



- Agents constraints: move in compass directions
- Adversary constraints: unknown, but its behavior is an SP2 language

Approach

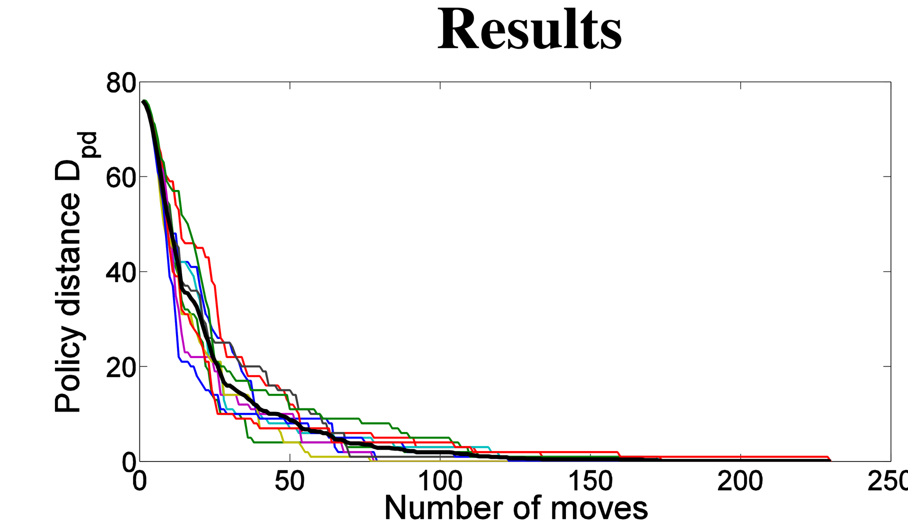


- Observe the environment and build a model for it
- Refine the model in real-time
- Use the refined model to update control strategies

Methodology

- Interaction between agents and environment takes the form of a deterministic zero-sum game
- On the game graph, progress toward satisfaction of the LTL spec is quantified
- Agents strategize assuming their hypothesis about the adversary plays its best move
- Control strategy is synthesized along standard model-checking approaches
- Adversary can move diagonally but not along compass more than once (agents do not know any of that at first)
- Agents' prior knowledge is that the environment behavior is in a specific class of formal languages
- They observe adversary actions and incrementally built a model for it
- The model is **guaranteed** to asymptotically converge to the true environment model
- After finite turns agents can recover the performance of full knowledge of their adversary dynamics

Results



In every game, as the environment model is refined and converges, the computed policy converges to the policy that would have been computed if environment dynamics were completely known.

Selected References:

[1] Kevin Leahy, Prasanna Kannappan, Adam Jardine, Jeffrey Heinz, Bert Tanner, and Calin Belta. Integration of Deterministic Inference with Formal Synthesis for Control under Uncertainty. Submitted to ACC 2015.
[2] J. Fu, H. G. Tanner, and J. Heinz,"Adaptive symbolic control for finite-state transition systems with grammatical inference," *In IEEE Trans. on Automatic Control*, 59(2):505-511, 2014.
[3] J. Fu, H. G. Tanner, and Jeffrey Heinz. Concurrent Multi-Agent Systems with Temporal Logic Objectives: Game Theoretic Analysis and Planning through Negotiation. *IET Control Theory and Applications* 9(3):465-474, 2014.
[4] J. Fu, H. G. Tanner, and J. Heinz, "Adaptive planning in unknown environments using grammatical inference" *In IEEE Conf. on Decision and Control*, 2013.
[5] Cristian Vasile and Calin Belta. Reactive Sampling-Based Temporal Logic Path Planning. In *International Conference on Robotics and Automation*, 2014.
[6] Xu Chu Ding, Mircea Lazar, and Calin Belta. Receding Horizon Temporal Logic Control for Finite Deterministic Systems. In *American Control Conference*, 2012.
[7] J. Rogers, J. Heinz, G. Bailey, M. Edlefsen, M. Visscher, D. Wellcome, and S. Wibel. "On Languages Piecewise Testable in the Strict Sense." *The Mathematics of Language* (2010):255-265
[8] J. Heinz. String Extension Learning. *In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, (2010):897-906

Supported by NSF CNS Under Awards
#103588
#103577