Learning and Inference in Dynamic Environments

Katsumi Inoue†
Principles of Informatics Research Division
National Institute of Informatics, Japan

† Inoue Lab members: Katsumi Inoue, Nicolas Schwind, Morgan Magnin*¹, Tenda Okimoto*², Tony Ribeiro*¹, Maxime Clement, Kotaro Okazaki, Taisuke Sato*³
(*¹: École Centrale de Nantes, *²: Kobe University, *³: AI Center, AIST)

NII IAB Meeting, October 28, 2015
Big, Changing, Continuous, Real-Time, and Incomplete Data
An internal model is constructed by learning from environment and interaction with other systems. The model is used for choosing the next action.
The effect/result of an action affects the environment and updates time-series data, history, experience and goals. Then the model is updated accordingly.
The agent also interacts with other agents, and its internal model is refined by such interactions.
Goal: Develop AI techniques that can identify the model of a system in dynamic environments and can achieve tasks even when unknown situations are encountered.
The Machine Learning Part constructs the model of the agent by abduction and induction from the goal, data, history and background knowledge.
Learning From Interpretation Transition (LFIT)

\[ \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \text{ for } t=0, t=1, \ldots, t=k-1, t=k, t=k+1, t=k+2, \ldots \]

**Input:** pairs of interpretations of the form \((I, J) \in 2^{HB} \times 2^{HB}\)

**Output:** logic program \(P\) such that \(J = T_P(I)\) for every \((I, J)\)

Learning Dynamical and Complex Networks

• Learning dynamic systems involving positive and negative feedbacks
• Learning Boolean networks from state transition diagrams
• Learning Cellular Automata from traces of configuration change


Learning Robust Boolean Networks


- Most transitions from $2^{12}$ states belong to the same *basin of attraction*.
- From this state transition, LFIT learned 54 state transition rules in 0.8 sec.
- An improved learning algorithm using BDD learned the same rules in 0.18 sec.

Given input \((S, T)\), a machine \(M\) produces an axiomatic system \(K\).

LFIT can learn meta-level one-step deduction rules, e.g., MP. The scenario can be applied to learning abduction and other non-standard logics.

Revising Plans in Adaptive Systems

Behavioural model learning/revision through **probabilistic rule learning**

Pathway Completion by Meta-Level Abduction

Cell cycle with cyclin-dependent kinases (Schneider et al., 2002)

Completing SBGN Networks with Gene Expression Data (collaboration with LRI/Paris-Sud & INRA-CNRS)

Biological phenomena

- Gene expression data
- Discretization
- (Discretized) observations

Formal system

- SBGN-AF network
- Causal network translation
- background theory

Inference engine

- Meta-Level Abduction
- Hypothesis
  - New interactions or predicates

Biological analysis

- Using database or literature

Hypothesis

- New interactions or predicates

The Decision Making Part chooses the best next action by solving constraint satisfaction problems (CSP) and constraint optimization problems (COP).
Dynamic Constraint Optimization Problems

• Most real life problems are dynamic, e.g., transportation, team formation, scheduling.
• Those models can be represented as (dynamic) (hard & soft) constraint networks.
• Goal: Minimize penalty and maximize reward.
• Requires fast computation of new solutions, yet some quality guarantees should be provided.
Dynamic COP: Applications and Approaches

- **Team formation**: Making robust teams of agents.
- **Nurse rerostering**: When a nurse is absent, build a new schedule with minimal and fair changes.
- **Timetabling**: Reconstruct timetables according to situation changes.

Two different approaches:
- **Proactive**: we prepare and take actions before changes happen.
- **Reactive**: we do not know what will happen and can only react to the changes.

Task Oriented Robust Team Formation

• **Goal:** Make a team to achieve a given set of tasks such that it can be effective even if some agents break down.

• A team is $k$-robust if it can still perform all tasks after losing any $k$ agents.

• Robustness is a proactive solution to prepare for unpredictable changes (the loss of some agents).

• TORTF does not prove more computationally demanding than the task-efficient team formation problem, i.e., robustness is *for free*.

Cyber Security Trade-Off Problem

• “You can't have 100 percent security and then also have 100 percent privacy and zero inconvenience.” (Barack Obama, 2013)
• Interception and communications data retention measures, even if the purpose is social security, are under the difficult trade-off between security, privacy and cost.”
• How to solve this trade-off and build the societal consensus?
• We used Multi-Objective Constraint Optimization (MO-COP) techniques.

Selected Solutions for Multi-Objective Problems

- Motivation: Product Configuration Example
  → Many (sometimes, thousands of) choices are given to the user.
  → Each alternative involves many criteria (e.g., for a car, price, lifetime, safety, brand reputation).
  → Impossible for users to choose their preferred product!

- Characterization of *representative solutions* (Figure b) / Efficient approximation procedures (Figure c)
- Interesting benefits for an iterative use, for large-scale problems.

The Model Checking Part verifies that the agent works correctly by predicting future results in dynamic environments.
Resilience of Constraint-Based Dynamic Systems

- Formalization of generic systems through COP
  - *variables*: components of the system
  - *constraints*: interactions between these components
  - *optimization function*: evaluates the quality of a configuration of the system

\[ \text{Resistance} + \text{Recoverability} + \text{Functionality} = \text{Resilience} \]

- Received The 3rd Prize of Best Challenges and Visions Papers.
Belief Propagation in Multi-Agent Systems

- We want to predict the propagation of fallacious beliefs in social networks (brand crisis management, e.g., Domino’s Pizza crisis in April 2009)
- We want to track the truth when several agents describe the same situation but have conflicting beliefs
Belief Propagation in Multi-Agent Systems

- Formalization of the framework, named "Belief Revision Games"
- Set of appealing properties:
  - 3 Preservation properties:
    - Convergence, Responsiveness, Monotonicity
  - 6 agents’ revision policies (ranging from credulous to skeptical ones)
- Investigation of the extent to which these properties are satisfied by the revision policies
- Robust framework, consistent with natural expectations:
  - credulous agents are more responsive than skeptical ones.
  - with skeptical agents, convergence is guaranteed.

Resilience Properties in Oscillatory Biological Systems

- Use parametric time Petri nets in order to analyze precisely the dynamic behavior of biological oscillatory systems, e.g., the mammalian circadian clock.
- Analyze resilience properties that endorse major changes in their environment, e.g., jet-lags, day-night alternating work-time.


Agent

Dynamic Environments

History/Experience

Goal

Background Knowledge

Decision Making

Model

Internal State

Time-Series Data

Machine Learning

Action

Dynamic Transition Model Checking

Other Agents

Dynamic Environments

\( B \land I \land H = O \)
Adaptation to IP Broadcasting Systems

- Background Knowledge Database [B]
- Viewer’s Journey Log [O]
- Machine Learning
- Target GRP [O]
- Rating Data [I]
- Exogenous Data

Dynamic Transition
Model Checking

AD Planning to each IP [H]

Internal State

Decision Making
AD Distribution
Contents Programming

Other Agents
Audience Reactions

Program/Content Audiences

Program/Content	Audiences

Log

Background

Knowledge

Database

Target

GRP

Machine	Learning

Decision
Making

Adapta8on
to	IP	Broadcas8ng	Systems
Inference and Learning in Dynamic Environments

AI techniques that can identify the model of a system in dynamic environments and can achieve tasks even when unknown situations are encountered.

Modeling
*(Machine Learning)*

Decision Making
*(Constraint Optimization)*

Prediction/Verification
*(Model Checking)*

Resilient Systems
Robustness/Sustainability

Dynamic Scheduling
Cyber Security

Opinion Construction
Multi-Agent Learning