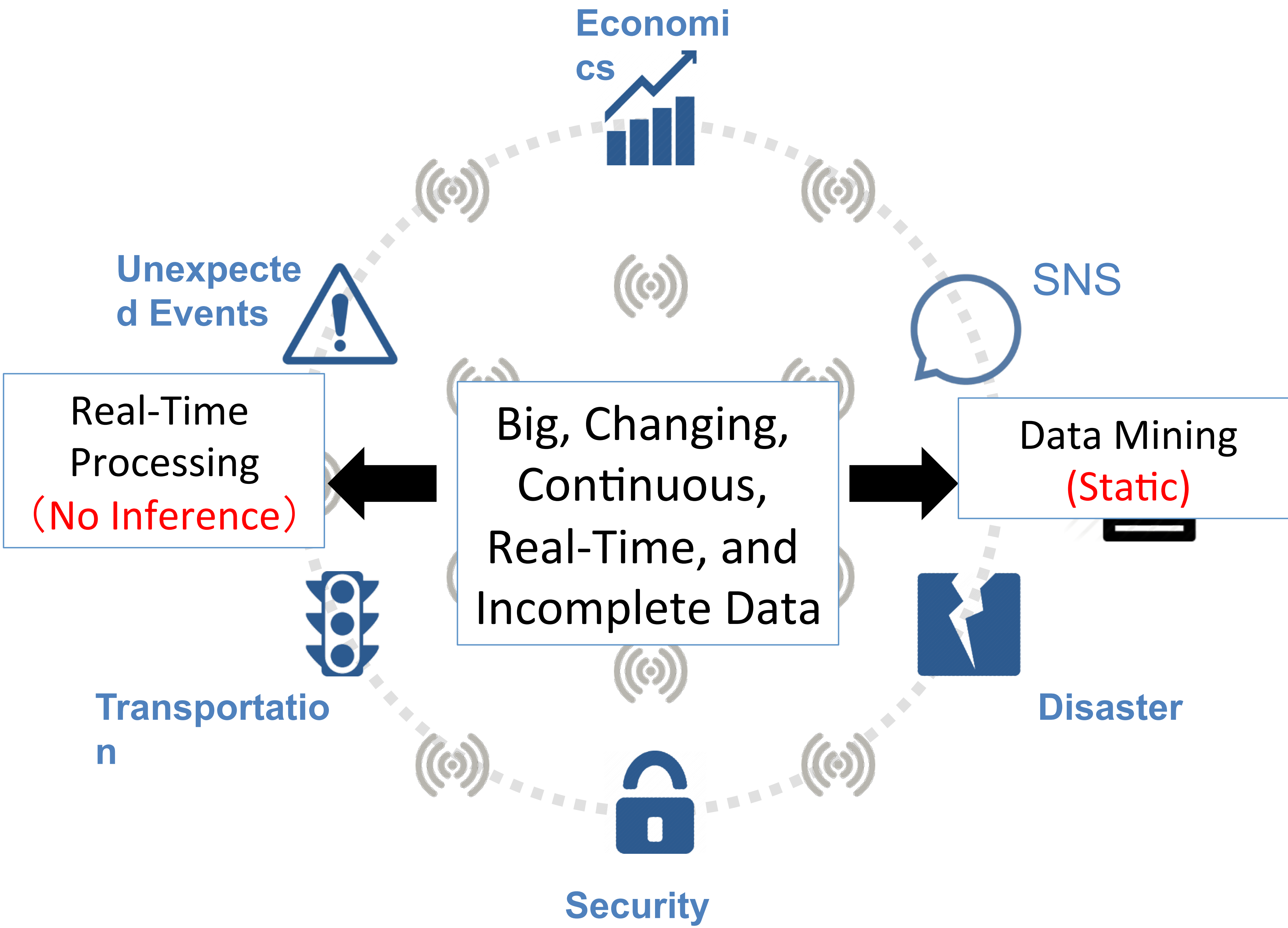
Learning and Inference in Dynamic Environments

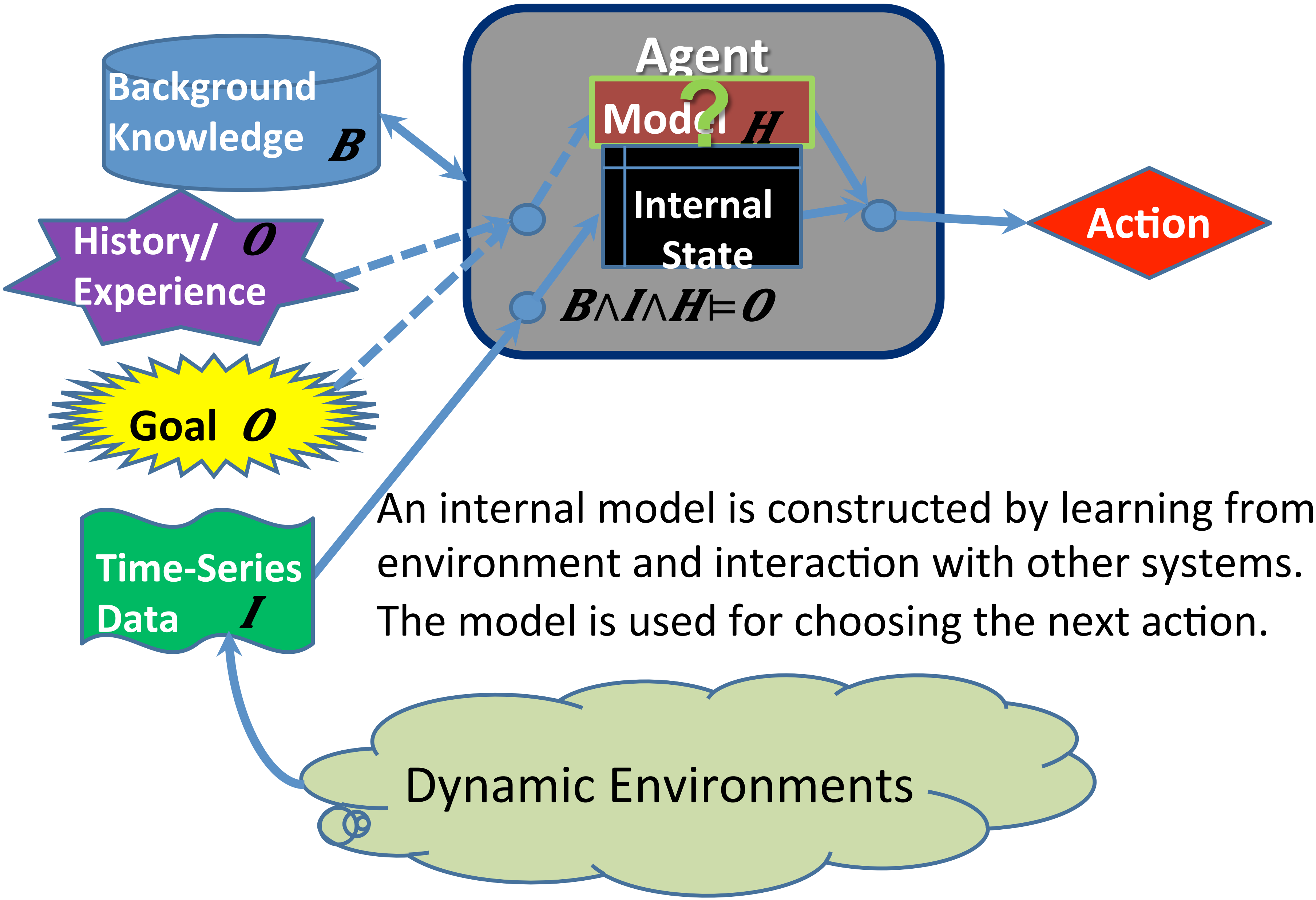
Katsumi Inoue†
Principles of Informatics Research Division
National Institute of Informatics, Japan

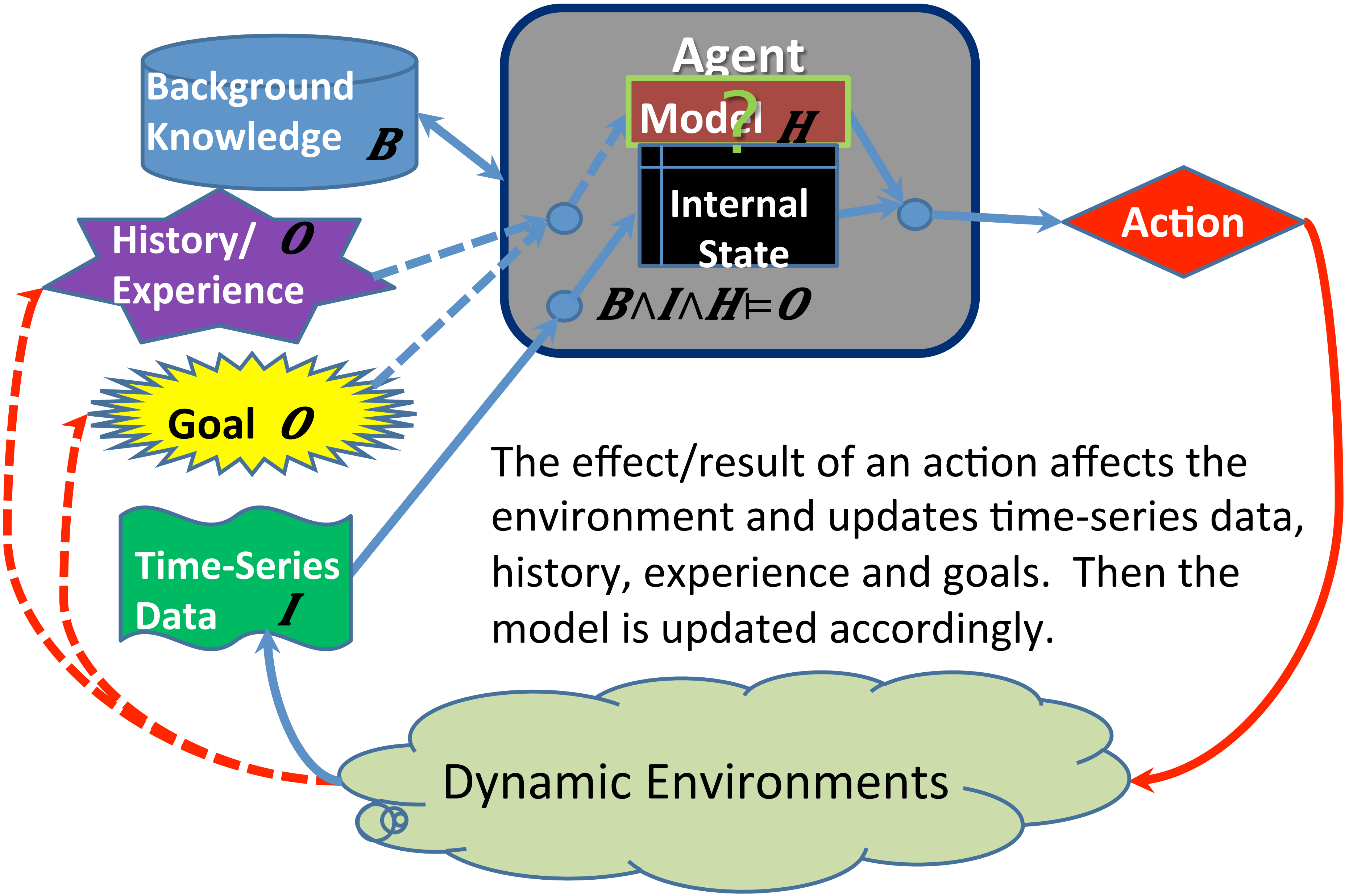
† **Inoue Lab members:** Katsumi Inoue, Nicolas Schwind, Morgan Magnin*[1], Tenda Okimoto*[2], Tony Ribeiro*[1], Maxime Clement, Kotaro Okazaki, Taisuke Sato*[3]

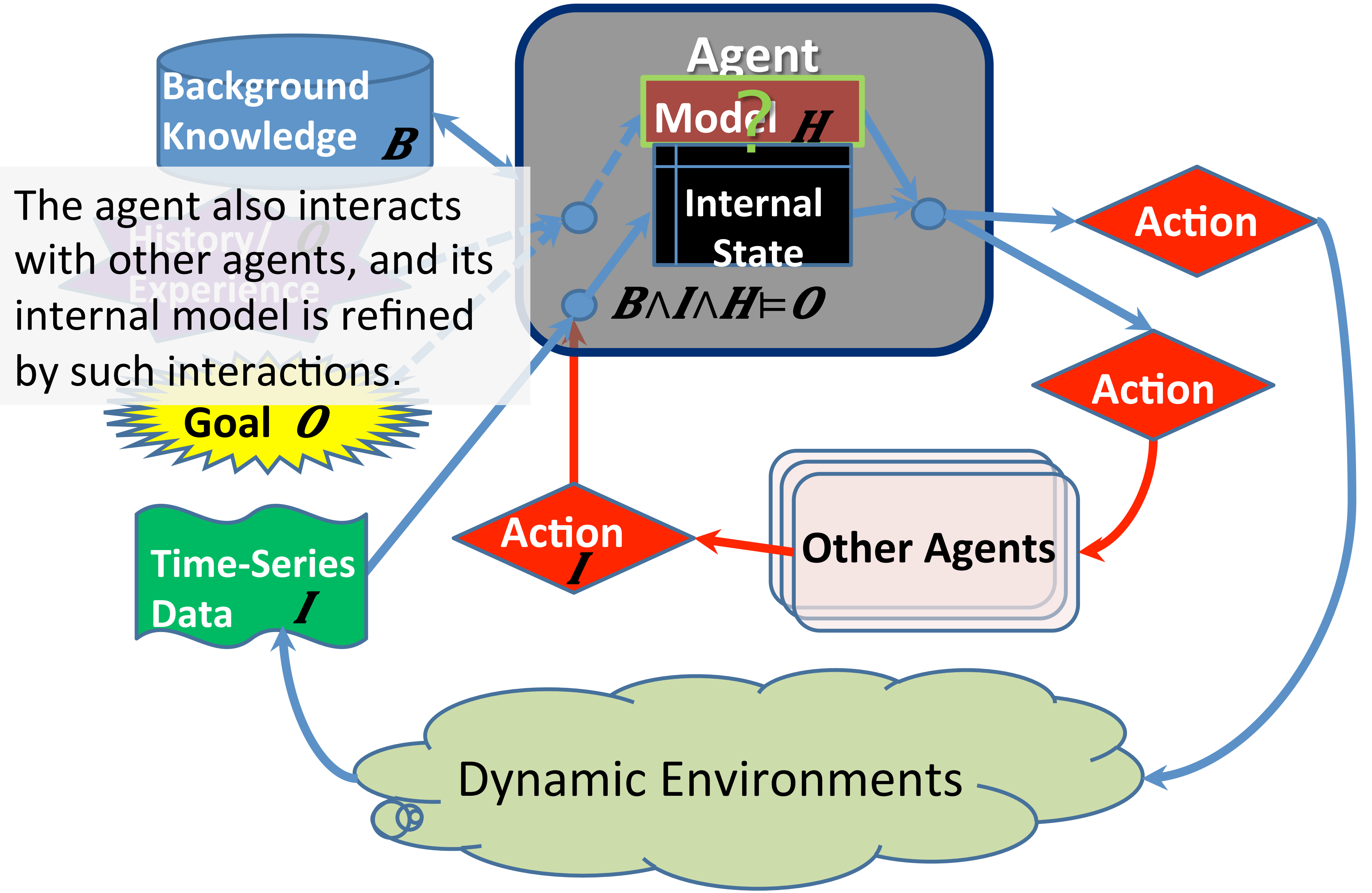
(*1: École Centrale de Nantes, *2: Kobe University, *3: AI Center, AIST)

***NII IAB Meeting**, October 28, 2015*

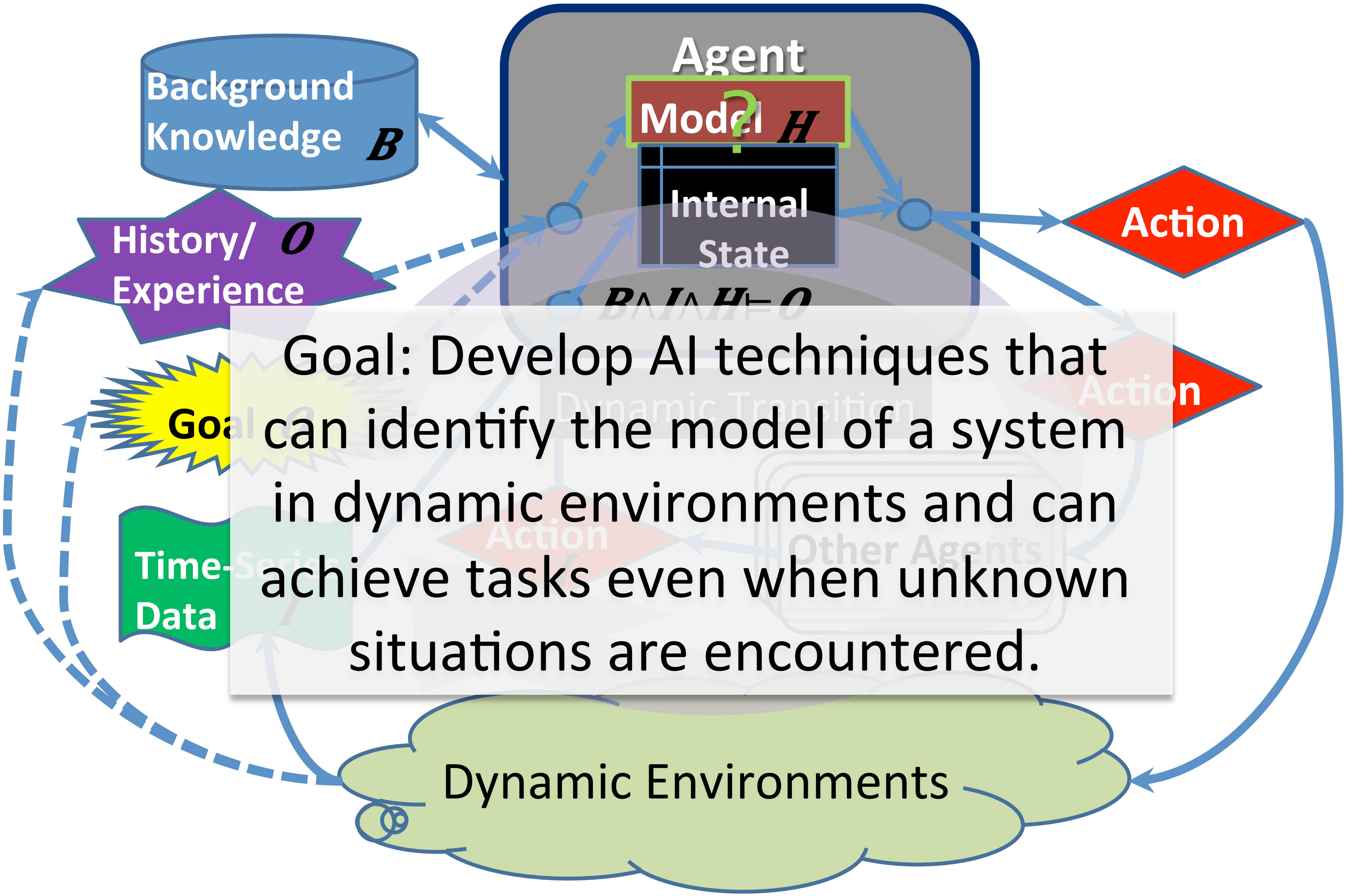

Economi
cs
Unexpecte
d Events
SNS
Real-Time
Processing
(No Inference)
Big, Changing,
Continuous,
Real-Time, and
Incomplete Data
Data Mining
(Static)
Transportatio
n
Disaster
Security

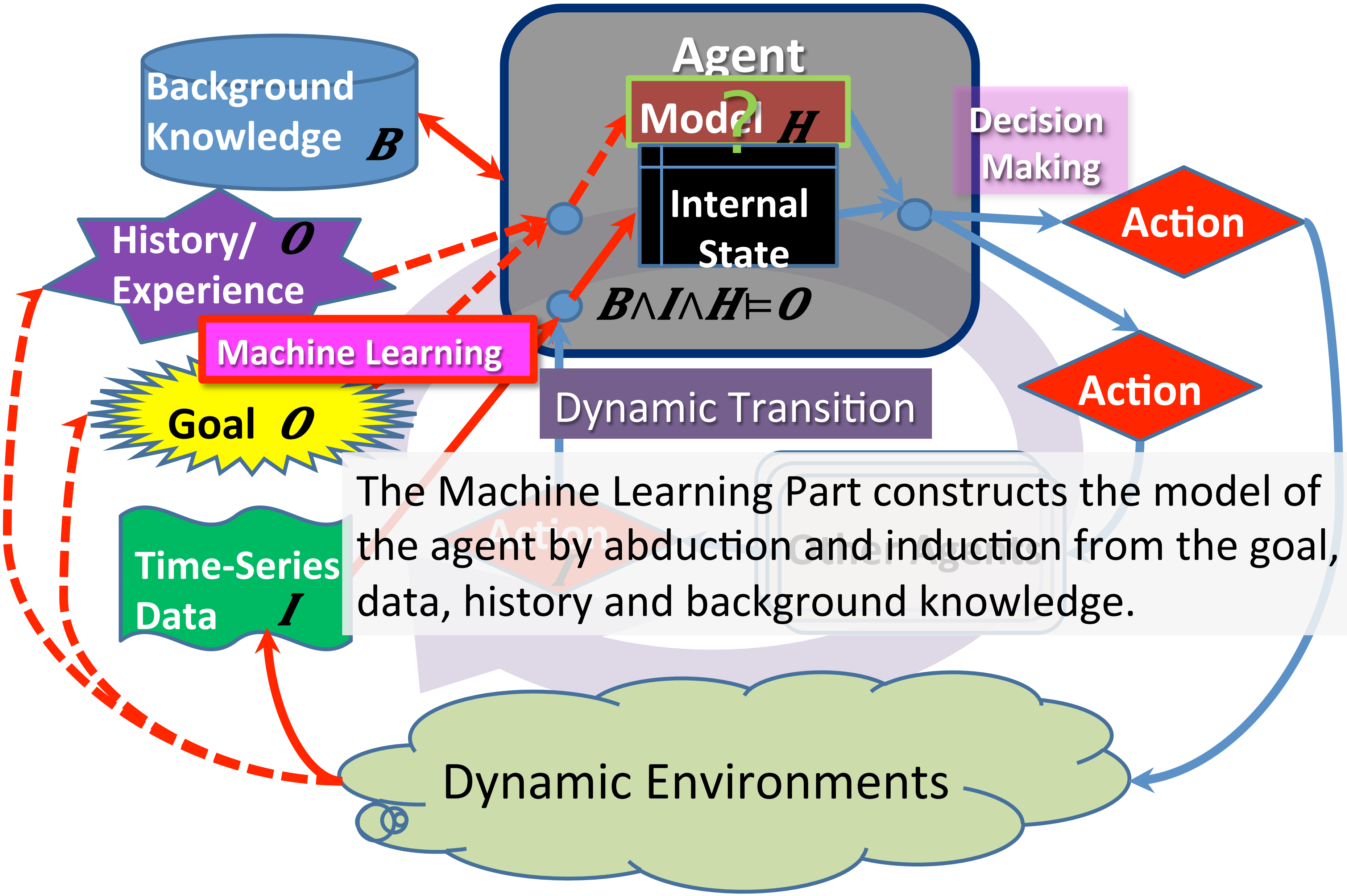

Background
Knowledge B
History/ O
Experience
Goal O
Time-Series
Data I
Agent
Model H
?
Internal
State
$B \wedge I \wedge H \models O$
Action
An internal model is constructed by learning from environment and interaction with other systems.
The model is used for choosing the next action.
Dynamic Environments


Agent
Model H
?
Internal State
B∧I∧H⊨O
Background Knowledge B
History/ Experience O
Goal O
Time-Series Data I
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.
Dynamic Environments



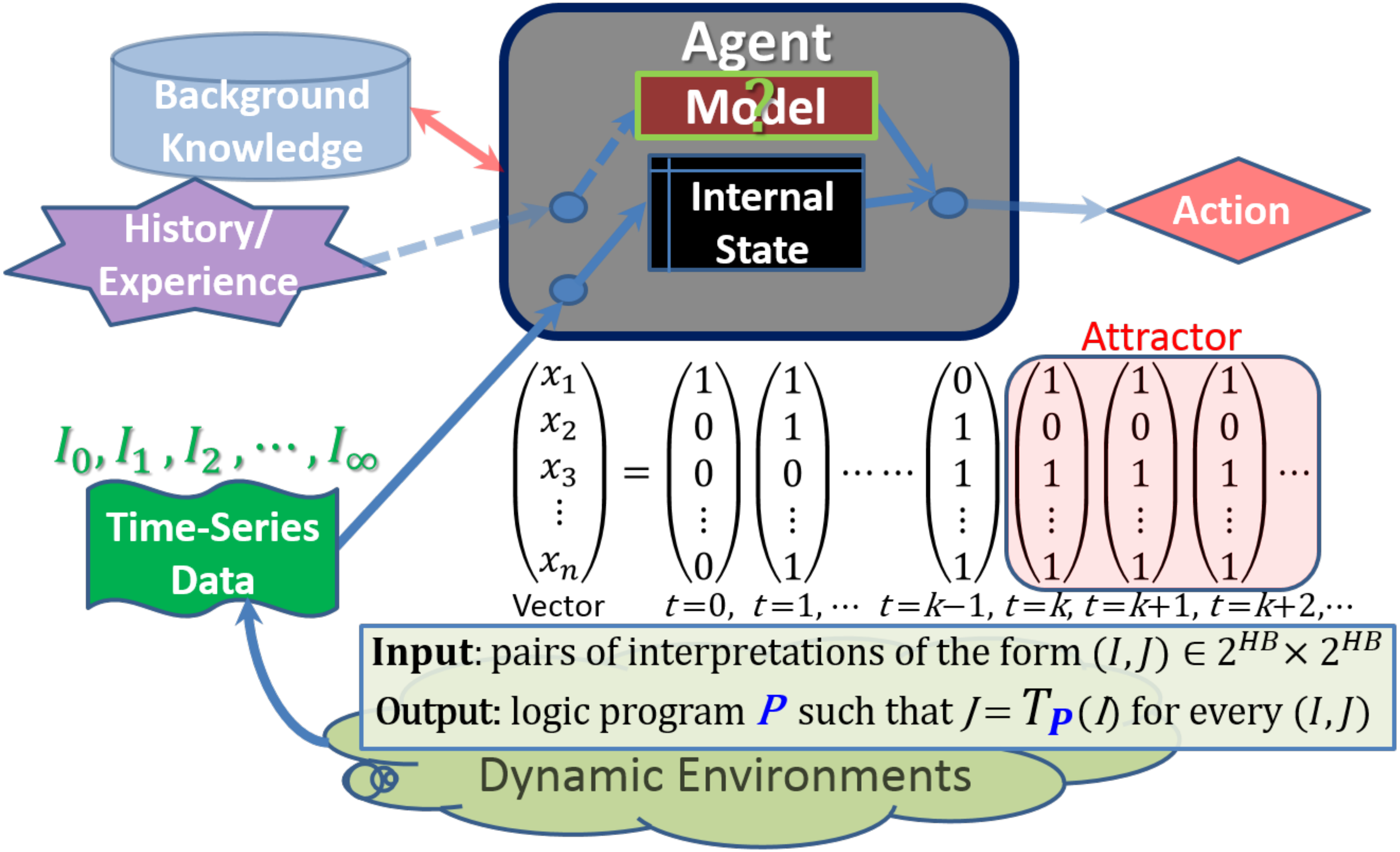
The agent also interacts with other agents, and its internal model is refined by such interactions.
Background Knowledge B
Agent
Model H
?
Internal State
B∧I∧H⊨O
History/ O
Experience
Goal O
Action
Action
Action I
Other Agents
Time-Series Data I
Dynamic Environments


Agent
Model ? H
Internal State
B∧I∧H⊨O
Background Knowledge B
History/ O Experience
Action
Action
Goal
Time-Series Data
Dynamic Transition
Action
Other Agents
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.
Dynamic Environments



The Machine Learning Part constructs the model of the agent by abduction and induction from the goal, data, history and background knowledge.

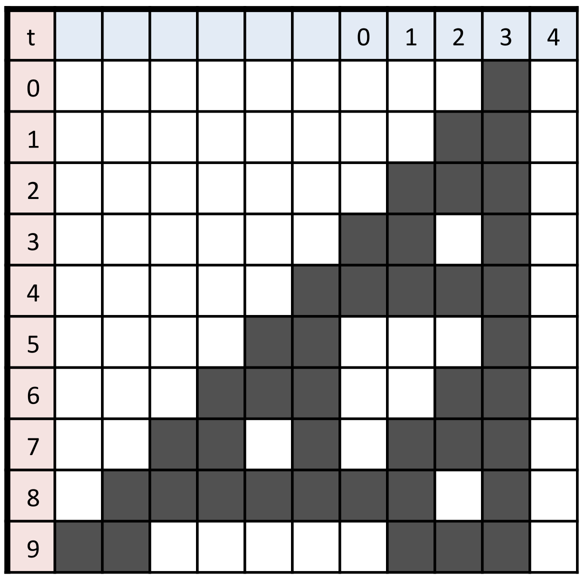
Learning From Interpretation Transition (LFIT)



- Inoue, K., Ribeiro, T., Sakama, C.: "Learning from Interpretation Transition", *Machine Learning*, 94(1):51-79, 2014.

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



- $c(x,t+1) \leftarrow c(x\text{-}1,t) \wedge c(x,t) \wedge \neg c(x+1,t).$
- $c(x,t+1) \leftarrow c(x\text{-}1,t) \wedge \neg c(x,t) \wedge c(x+1,t).$
- $c(x,t+1) \leftarrow \neg c(x\text{-}1,t) \wedge c(x,t) \wedge c(x+1,t).$
- $c(x,t+1) \leftarrow \neg c(x\text{-}1,t) \wedge c(x,t) \wedge \neg c(x+1,t).$
- $c(x,t+1) \leftarrow \neg c(x\text{-}1,t) \wedge \neg c(x,t) \wedge c(x+1,t).$

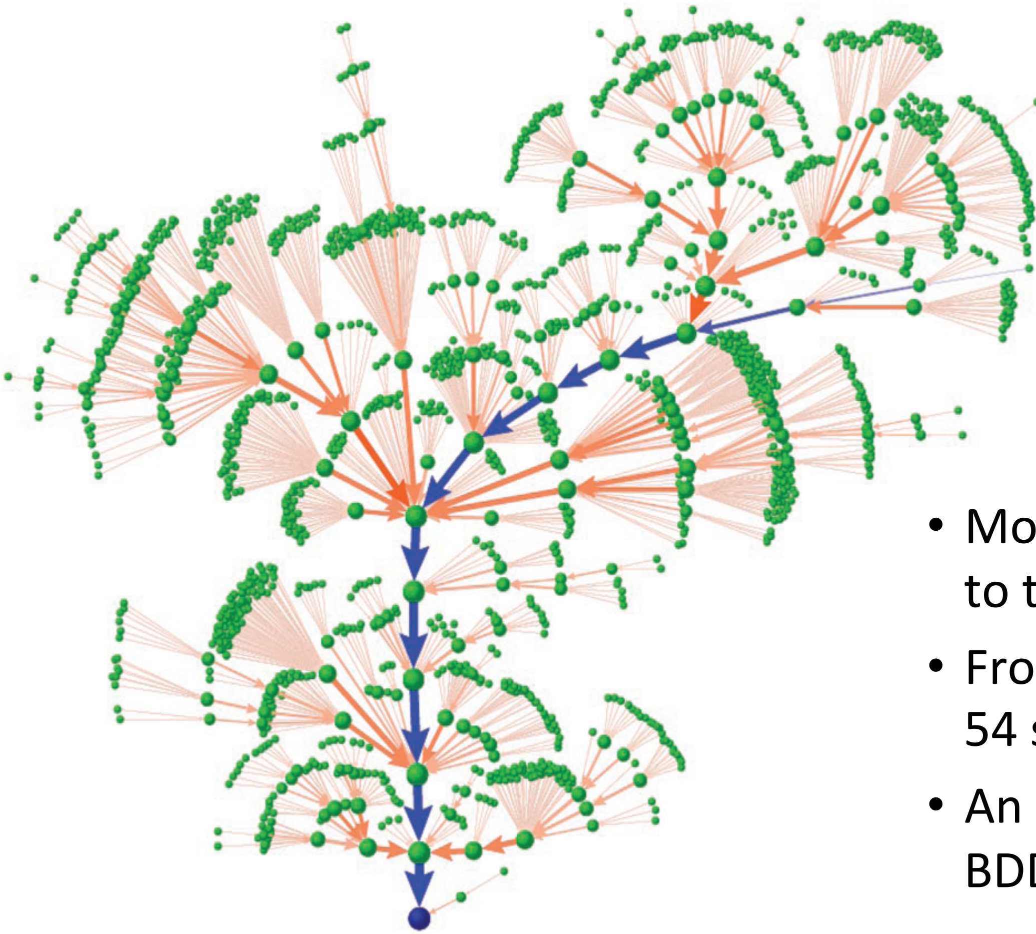
current pattern	111	110	101	100	011	010	001	000
new state for center cell	0	1	1	0	1	1	1	0

Wolfram's Rule 110 (Turing-complete)

➢ Inoue, K., Ribeiro, T., Sakama, C.: "Learning from Interpretation Transition", *Machine Learning,* 94(1):51-79, 2014.

➢ Völker, M., Inoue, K.: "Logic Programming for Cellular Automata", *ICLP 2015*.

Learning Robust Boolean Networks

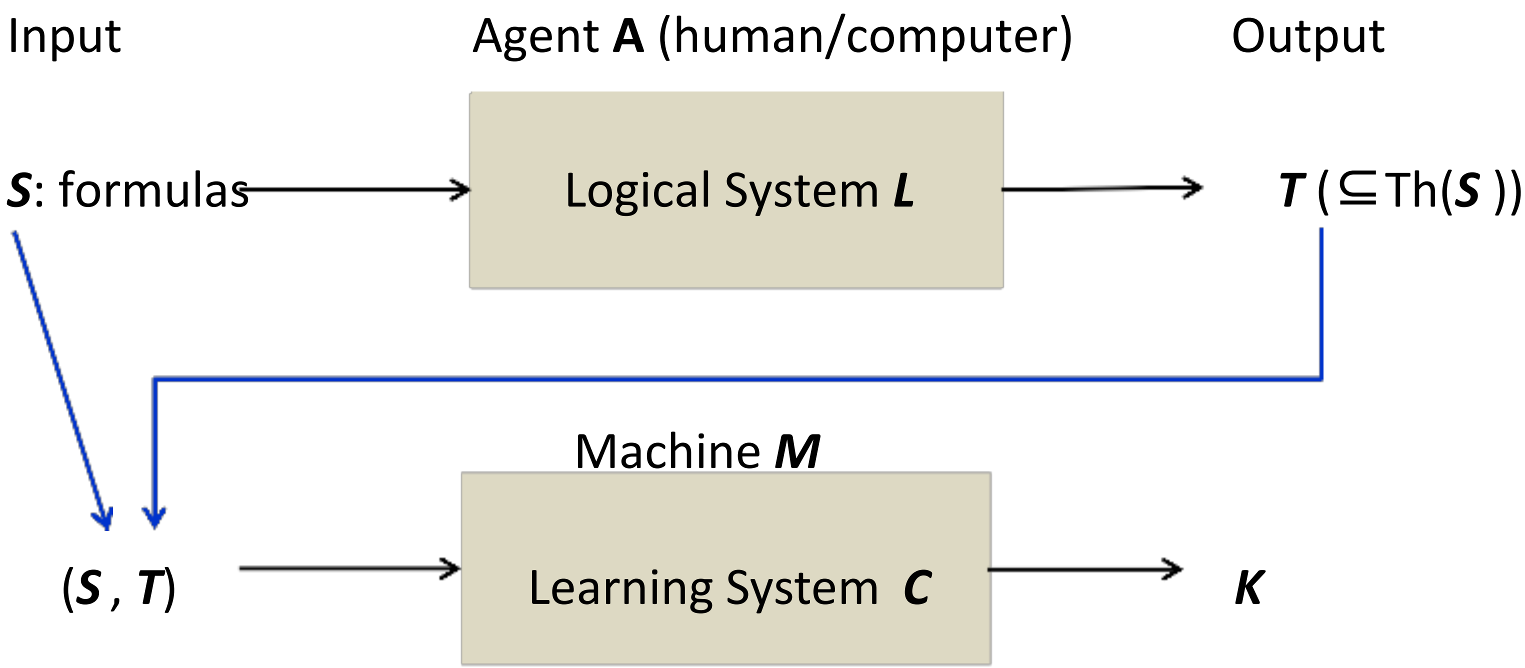


Li, F. *et al*.: The yeast cell-cycle network is robustly designed, *PNAS,* 101(14), 2004.

- 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.

- Inoue, K., Ribeiro T., Sakama, C.: "Learning from Interpretation Transition", *Machine Learning,* 94(1):51-79, 2014.
- Ribeiro, T., Inoue, K., Sakama, C.: "A BDD-Based Algorithm for Learning from Interpretation Transition", *Post-Proc. ILP 2013*, *LNAI*, Vol.8812, pp.47-63, 2014.

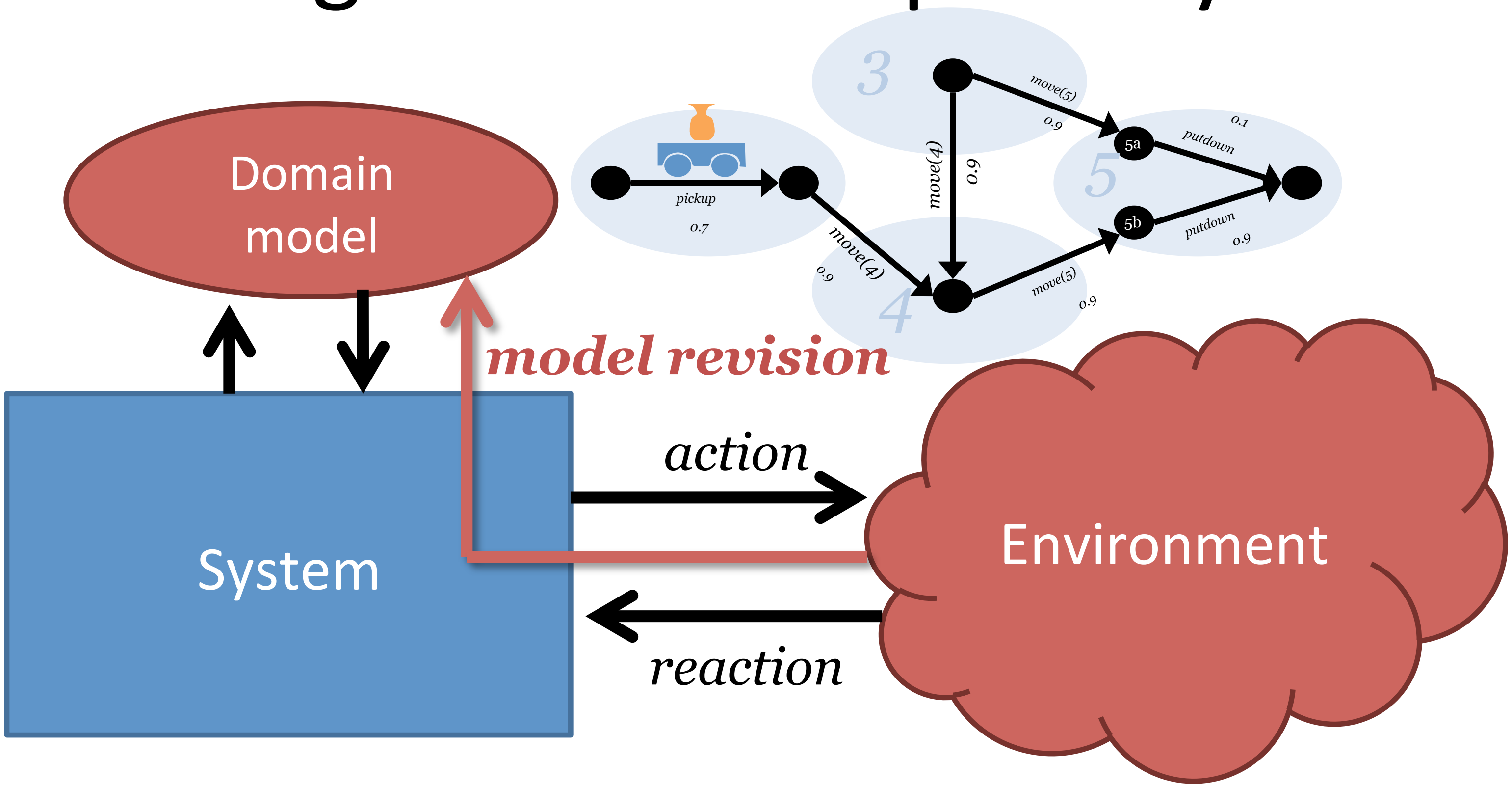
"Can Machines Learn Logics?"



- 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.

➢ Sakama, C., Inoue, K.: "Can Machines Learn Logics?", *AGI 2015*, LNAI, Vol.9205, pp. 341-351, 2015.

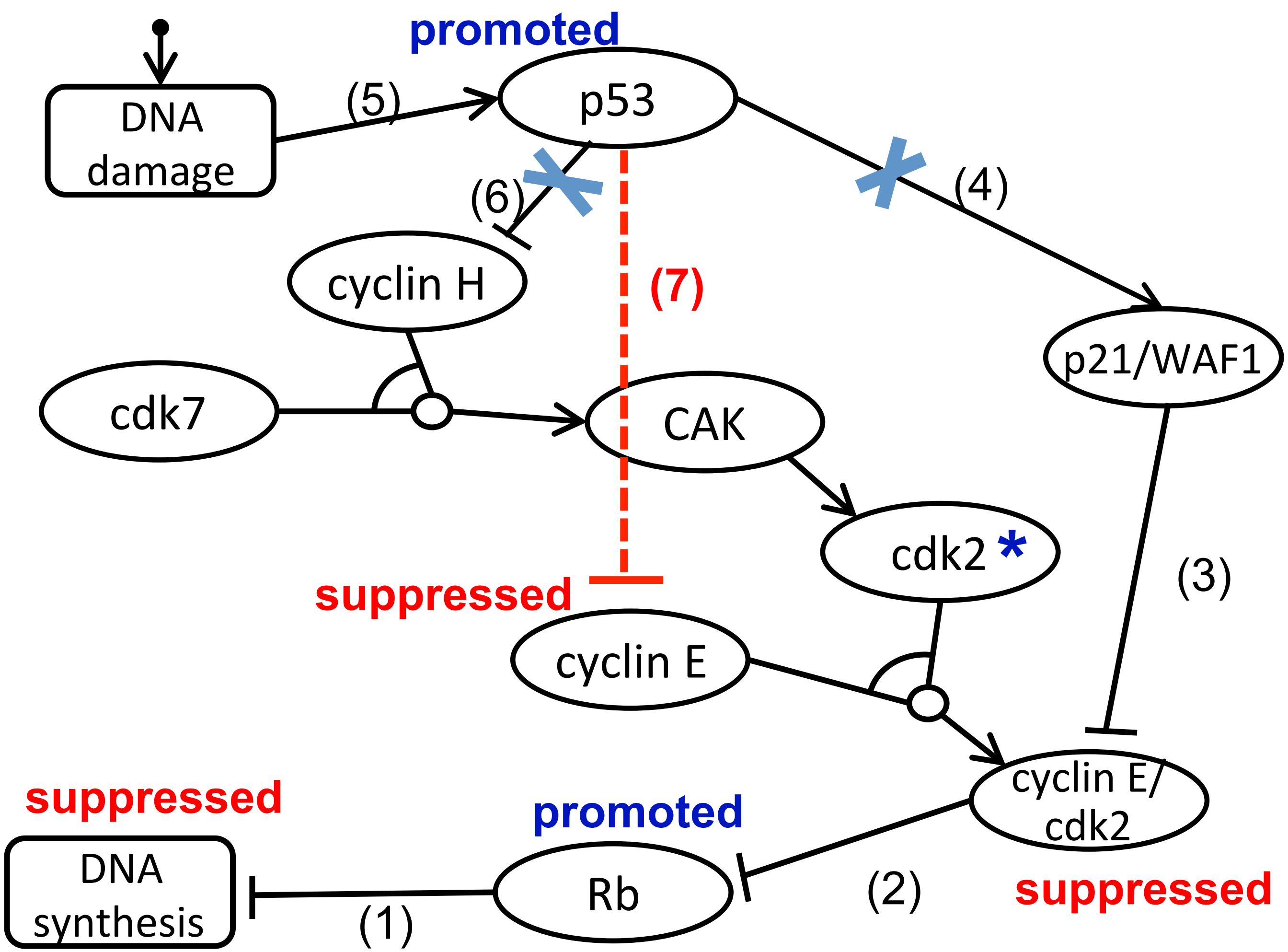
Revising Plans in Adaptive Systems



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

- Sykes, D., Corapi, D., Magee, J., Kramer, J., Russo, A., Inoue, K.: "Learning revised models for planning in adaptive systems", *ICSE 2013*: 63-71.
- Martínez, D., Ribeiro, T., Inoue, K., Alenyà, G., Torras, C.: "Learning Probabilistic Action Models from Interpretation Transitions", *ICLP 2015*.

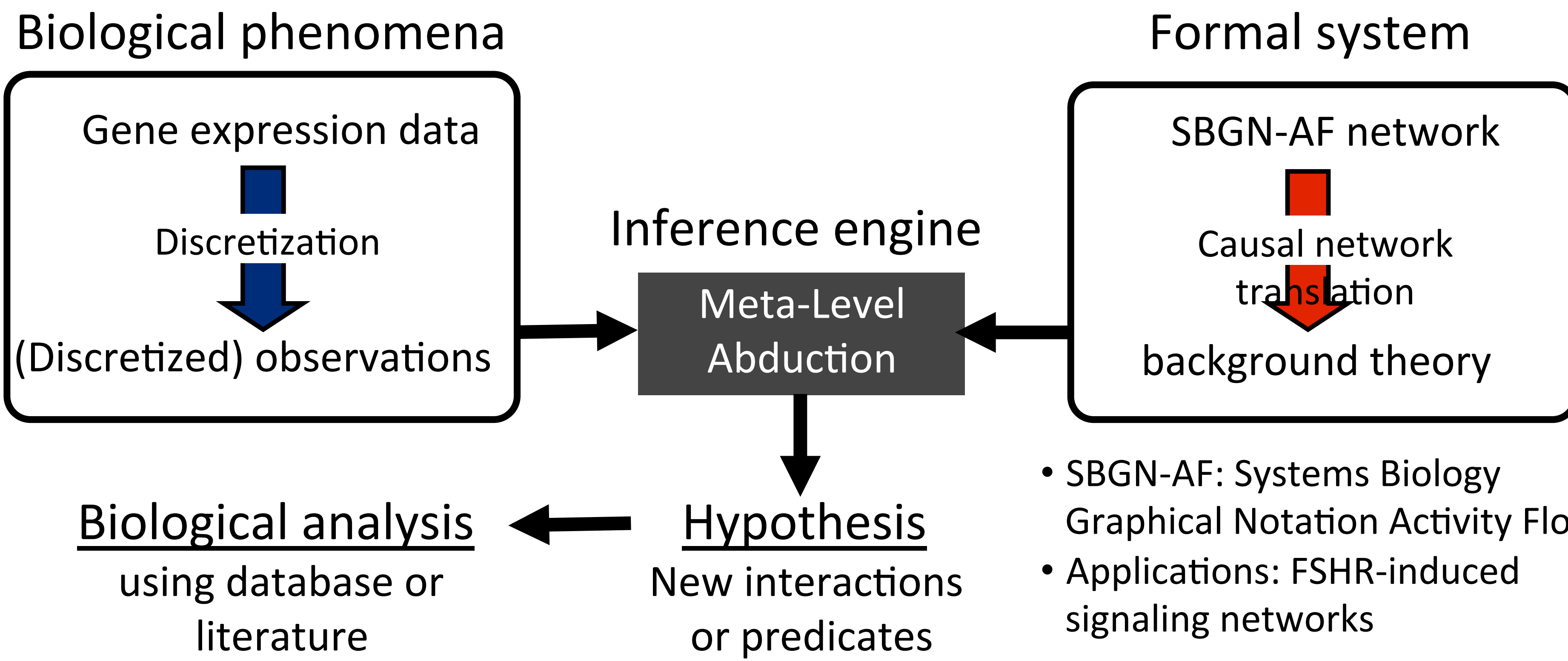
Pathway Completion by Meta-Level Abduction



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

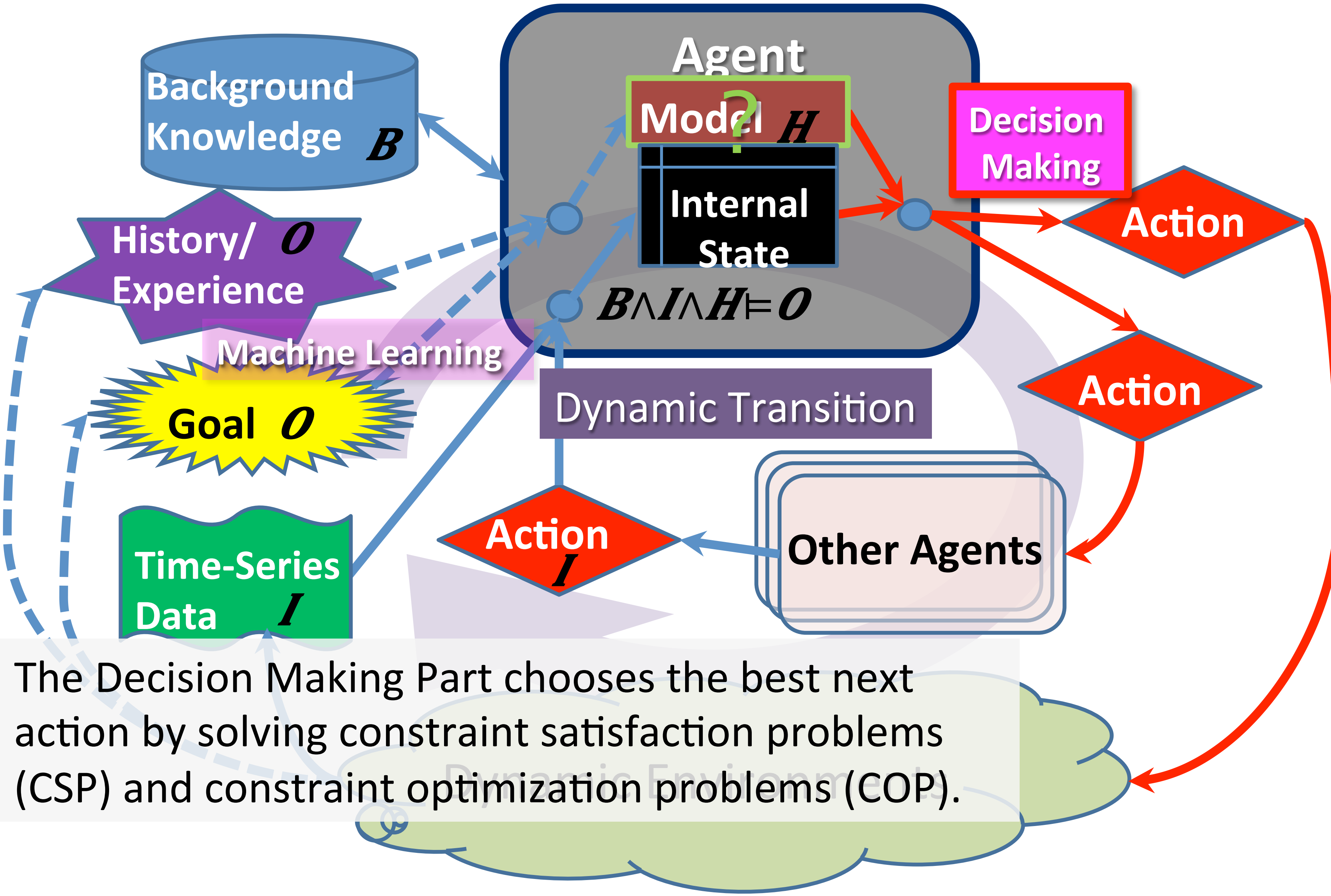
- Inoue, K., Doncescu, A., Nabeshima, H.: "Completing causal networks by meta-level abduction, *Machine Learning*, 91(2):239-277, 2013.

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



- SBGN-AF: Systems Biology Graphical Notation Activity Flow
- Applications: FSHR-induced signaling networks

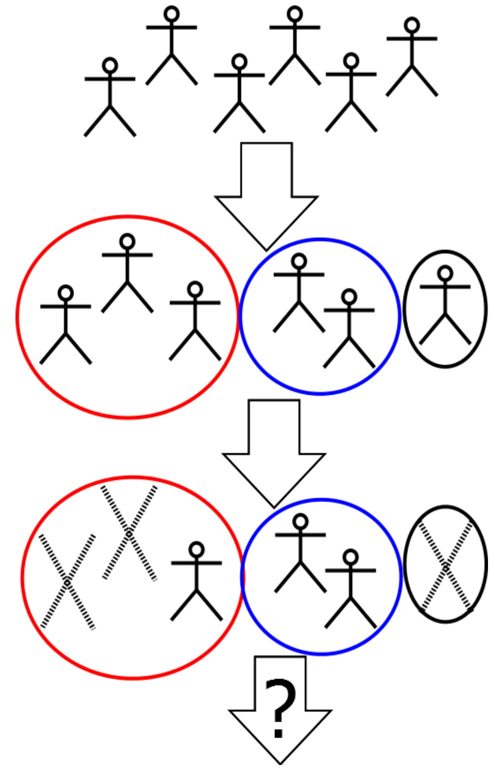
➢ Yamamoto, Y., Rougny, A., Nabeshima, H., Inoue, K., Moriya, H., Froidevaux, C., Iwanuma, K.: "Completing SBGN-AF Networks by Logic-Based Hypothesis Finding", *FMMB 2014*, LNBI, Vol.8738, pp.165-179, 2014.

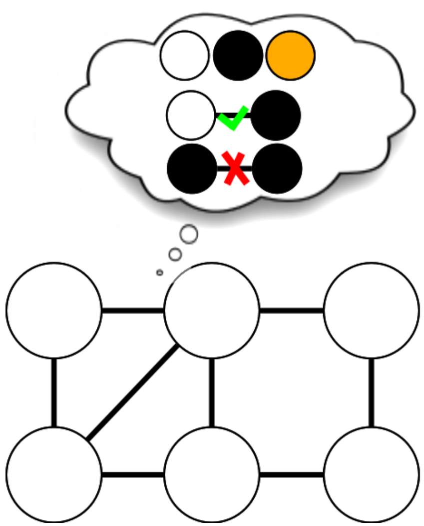


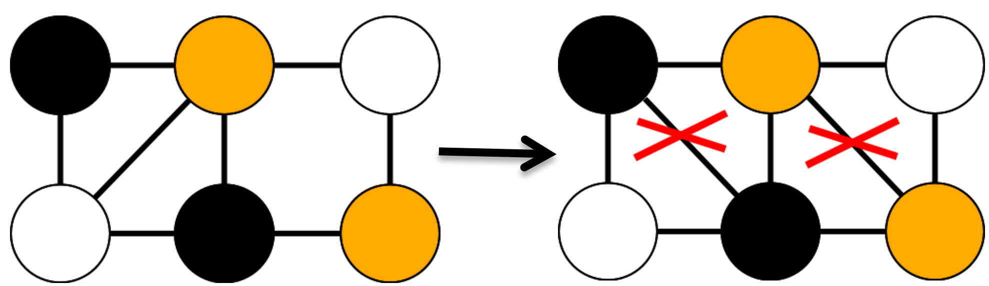
The Decision Making Part chooses the best next action by solving constraint satisfaction problems (CSP) and constraint optimization problems (COP).

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.





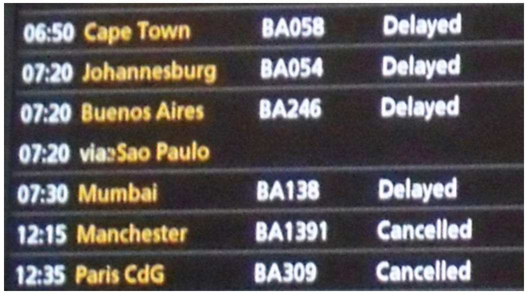


Dynamic COP: Applications and Approaches

- *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.

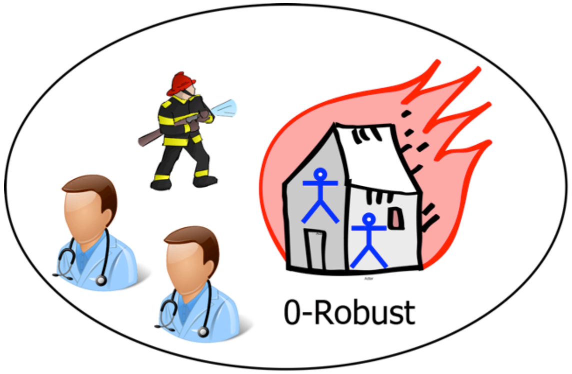


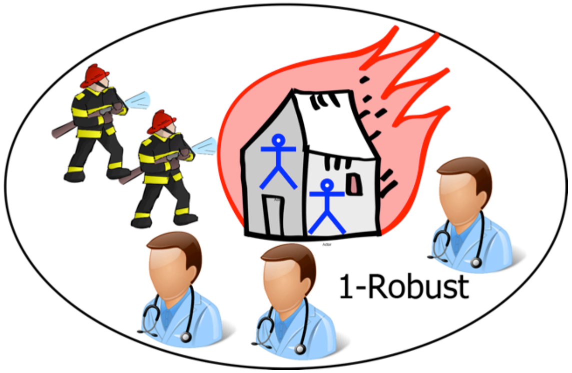
➢ Okimoto, T., Schwind, N., Clement, M., Ribeiro, T., Inoue, K., Marquis, P.: "How to Form a Task-Oriented Robust Team", *AAMAS 2015*: 395-403.

➢ Clement, M., Okimoto, T., Schwind, N., Inoue, K.: "Finding Resilient Solutions for Dynamic Multi-Objective Constraint Optimization Problems", *ICAART 2015*: 509-516.

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*.





➢ Okimoto, T., Schwind, N., Clement, M., Ribeiro, T., Inoue, K., Marquis, P.: "How to Form a Task-Oriented Robust Team", *AAMAS 2015*: 395-403.

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 <u>security</u>, <u>privacy</u> and <u>cost</u>."
- How to solve this trade-off and build the societal consensus?
- We used *Multi-Objective Constraint Optimization* (MO-COP) techniques.

➢ Okimoto, T., Ikegai, N., Ribeiro, T., Inoue, K., Okada, H., Maruyama, H.: Cyber Security Problem Based on Multi-Objective Distributed Constraint Optimization Technique, *WSR 2013*.

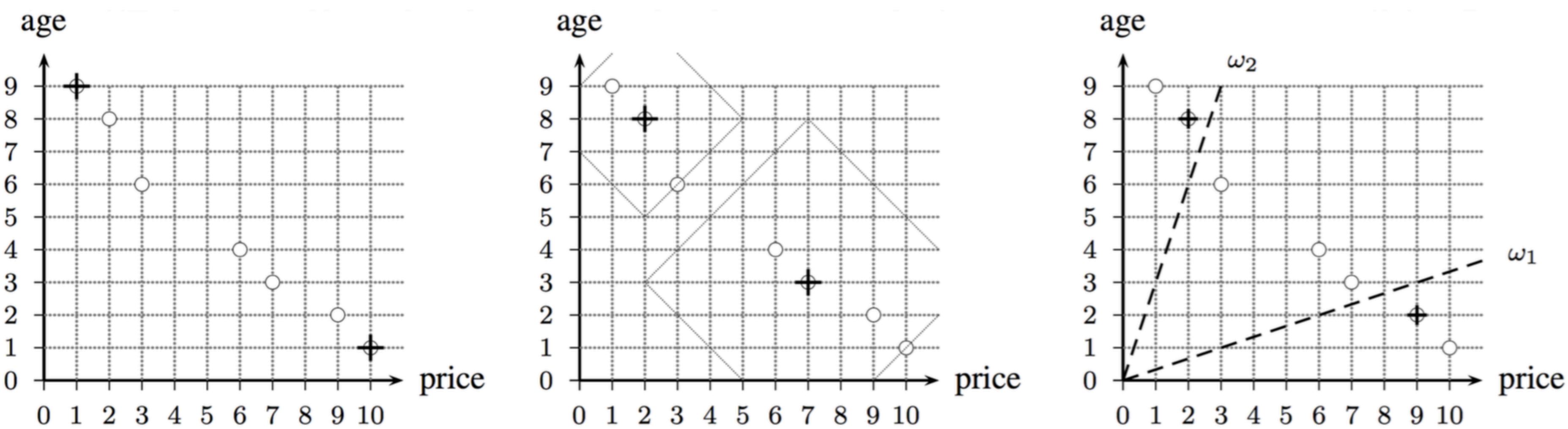
Selected Solutions for Multi-Objective Problems

- 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).

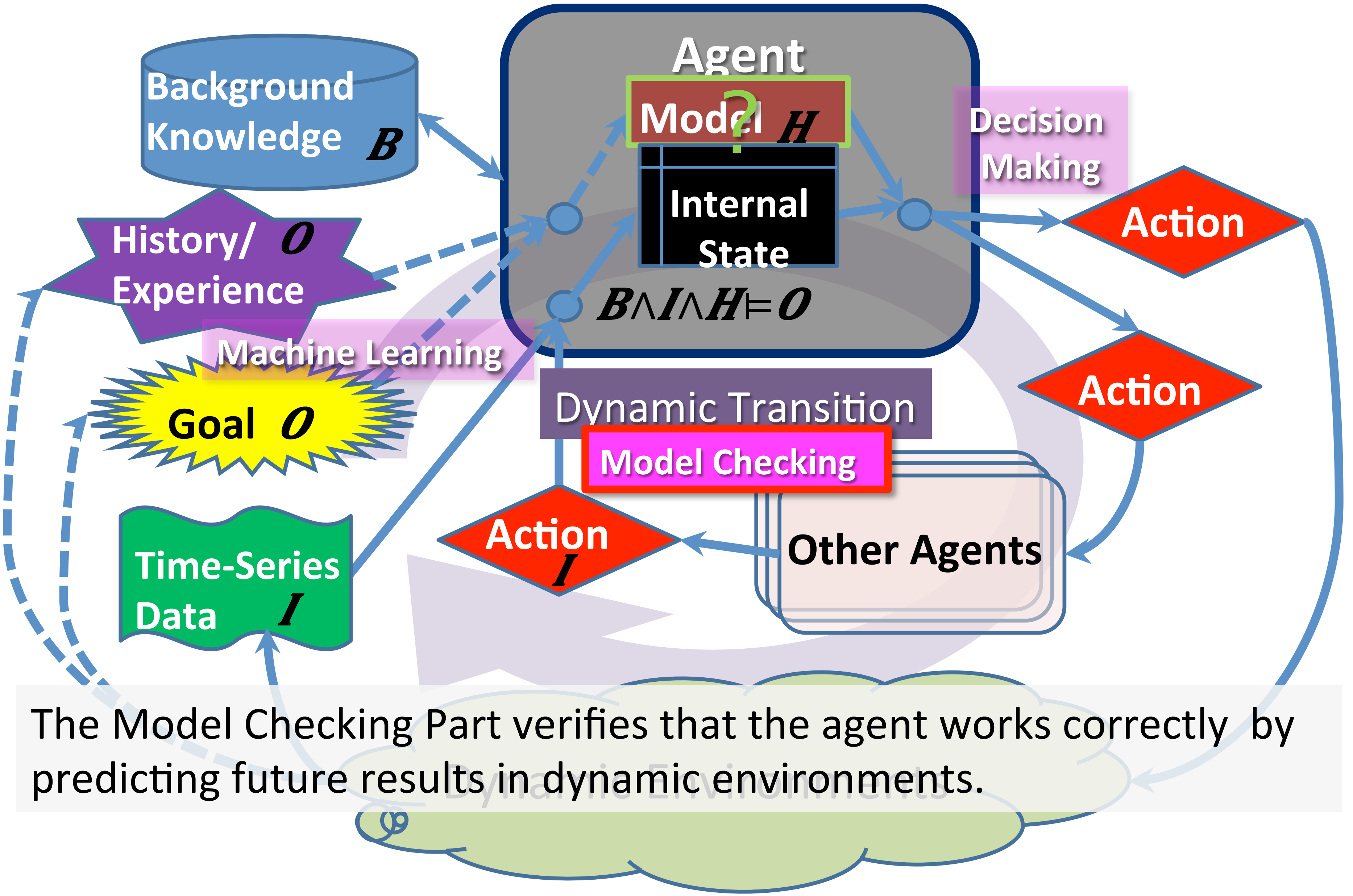
  → Impossible for users to choose their preferred product!



(a) "Diverse" solutions (existing works) (b) Our proposal (c) Our proposal (efficient approximation)

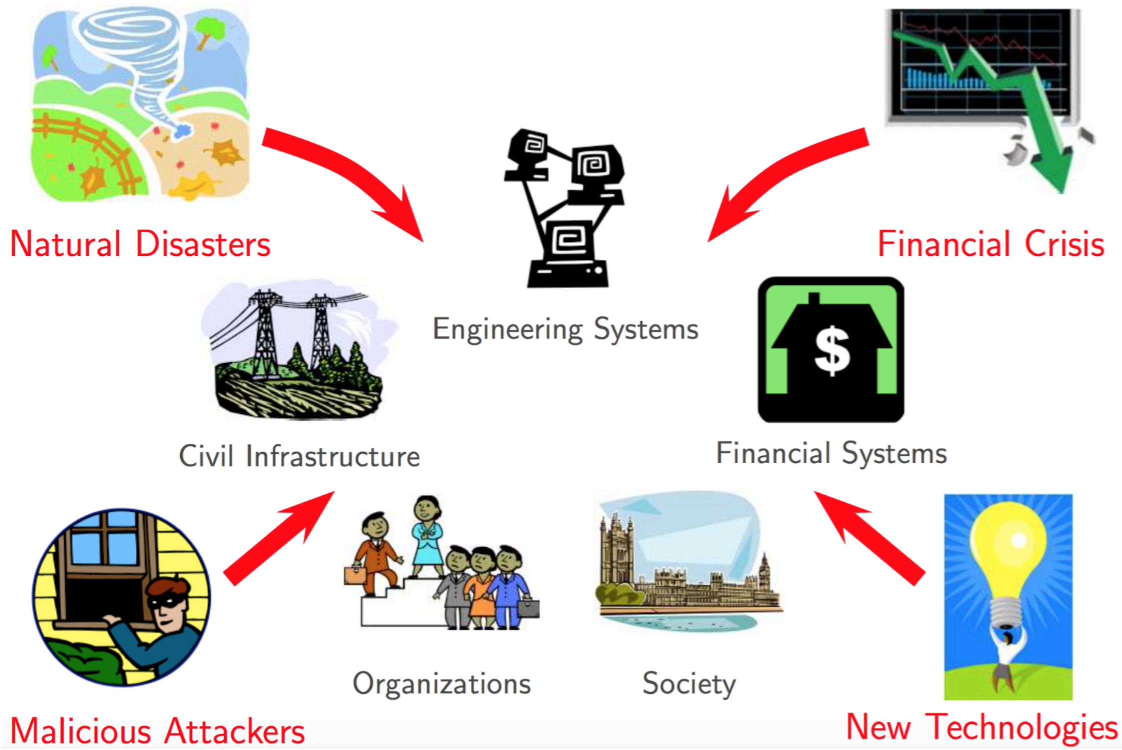
- Characterization of *representative solutions* (Figure b) / Efficient approximation procedures (Figure c)
- Interesting benefits for an iterative use, for large-scale problems.

➢ Schwind, N., Okimoto, T., Konieczny, S., M., Inoue, K.: to appear, 2016.



The Model Checking Part verifies that the agent works correctly by predicting future results in dynamic environments.

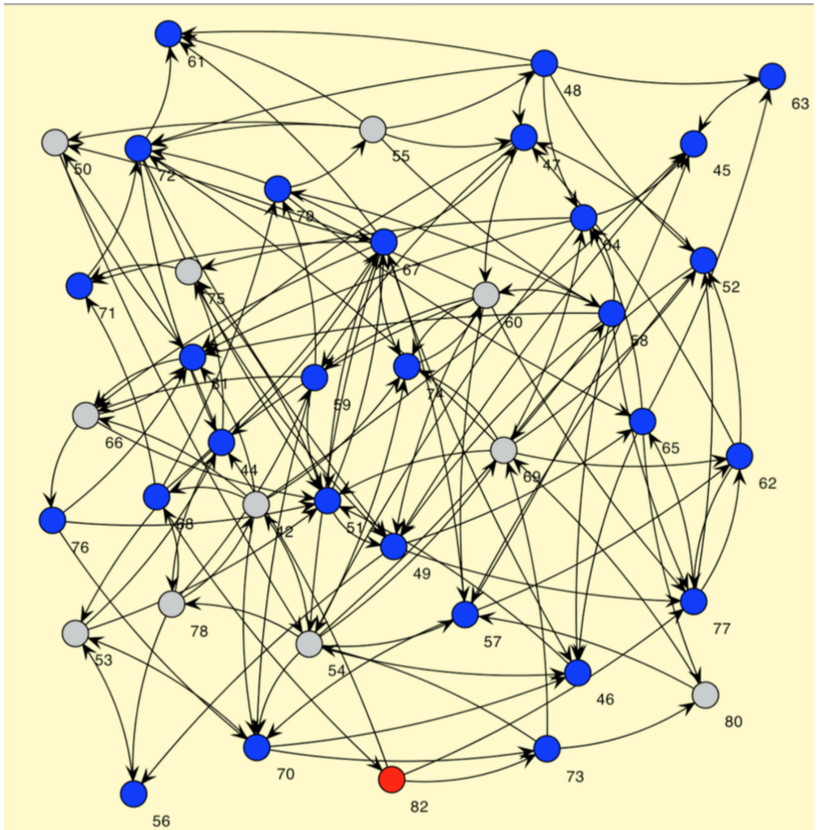
Resilience of Constraint-Based Dynamic Systems

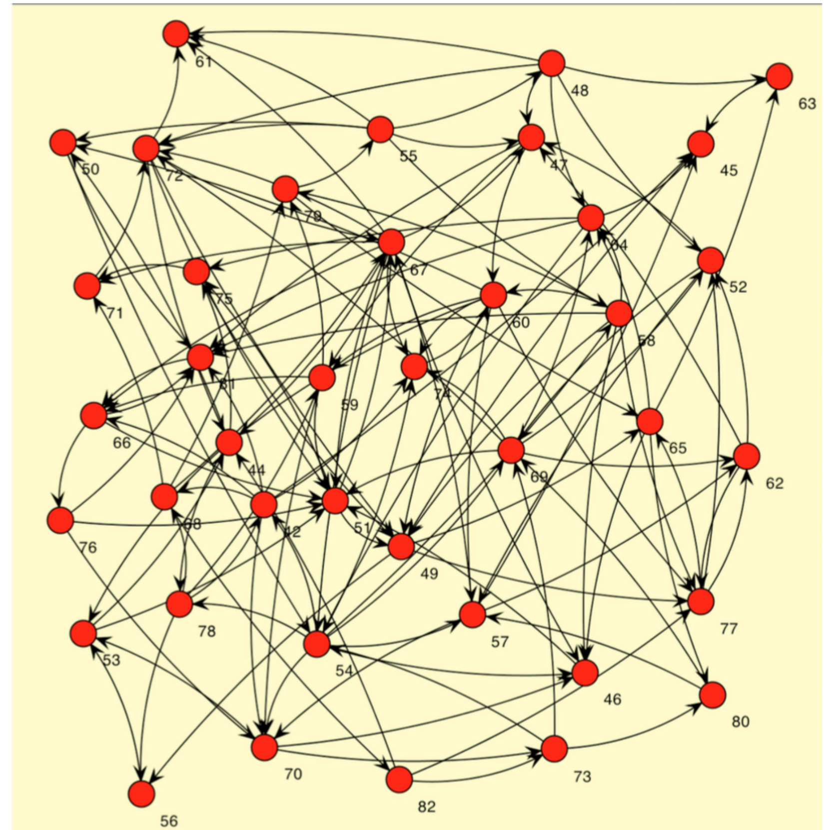


- Formalization of generic systems through COP
  - *variables*: components of the system
  - *constraints*: interactions between these components
  - *optimization function*: evaluates the quality of a configuration of the system
- Resistance + Recoverability + Functionality = **Resilience**

  - Schwind, N., Okimoto, T., Inoue, K., Chan, H., Ribeiro, T., Minami, K., Maruyama, H.: "Systems Resilience: A Challenge Problem for Dynamic Constraint-Based Agent Systems", *AAMAS 2013*, pp.785-788.
  - 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:

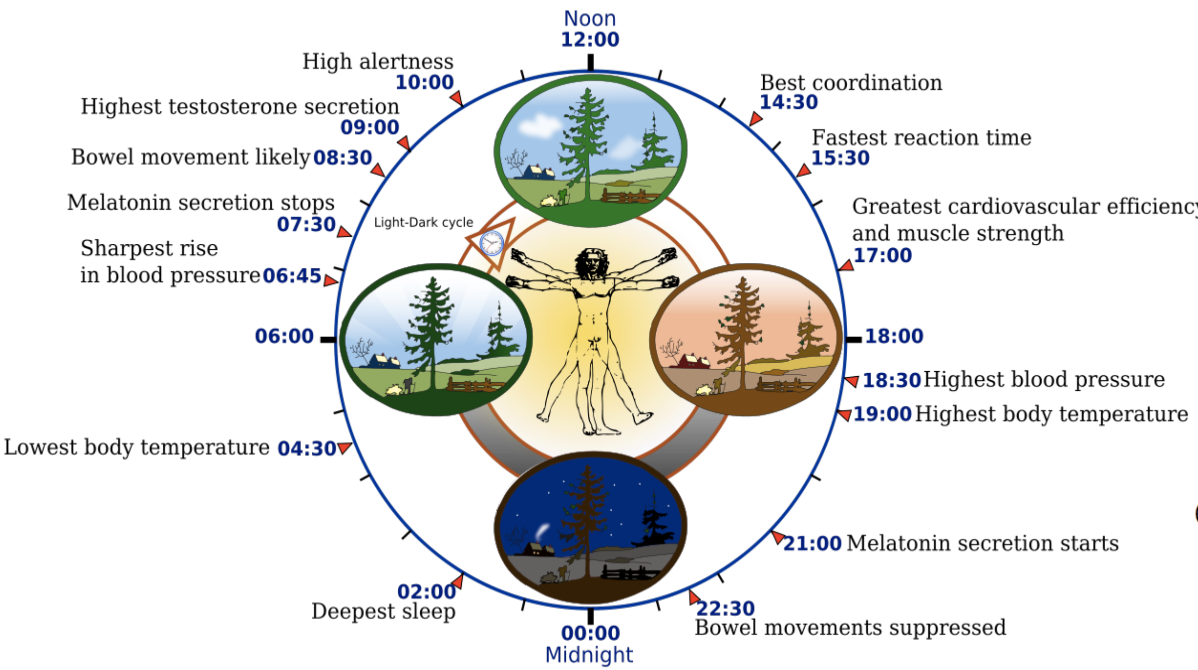
    Convergence, Responsiveness, Monotonicity
  - 6 agents' revision policies (ranging from credulous to skeptical ones)
- Investigation of the extent to which these properties are satisfied by the revision policies
- Robust framework, consistent with natural expectations:

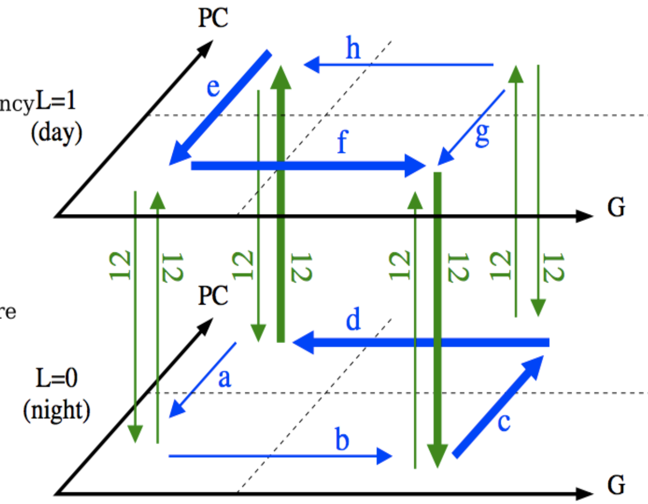
  → credulous agents are more responsive than skeptical ones.

  → with skeptical agents, convergence is guaranteed.

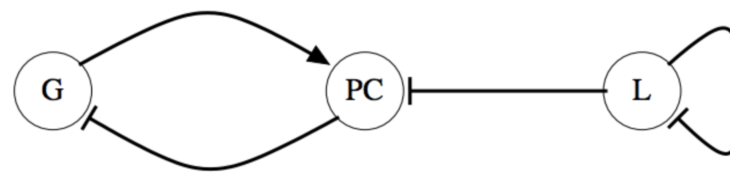
➢ Schwind, N., Inoue, K., Bourgne, G., Konieczny, S., Marquis, P.: "Belief Revision Games", *AAAI-15*: 1590-1596, 2015.

Resilience Properties in Oscillatory Biological Systems



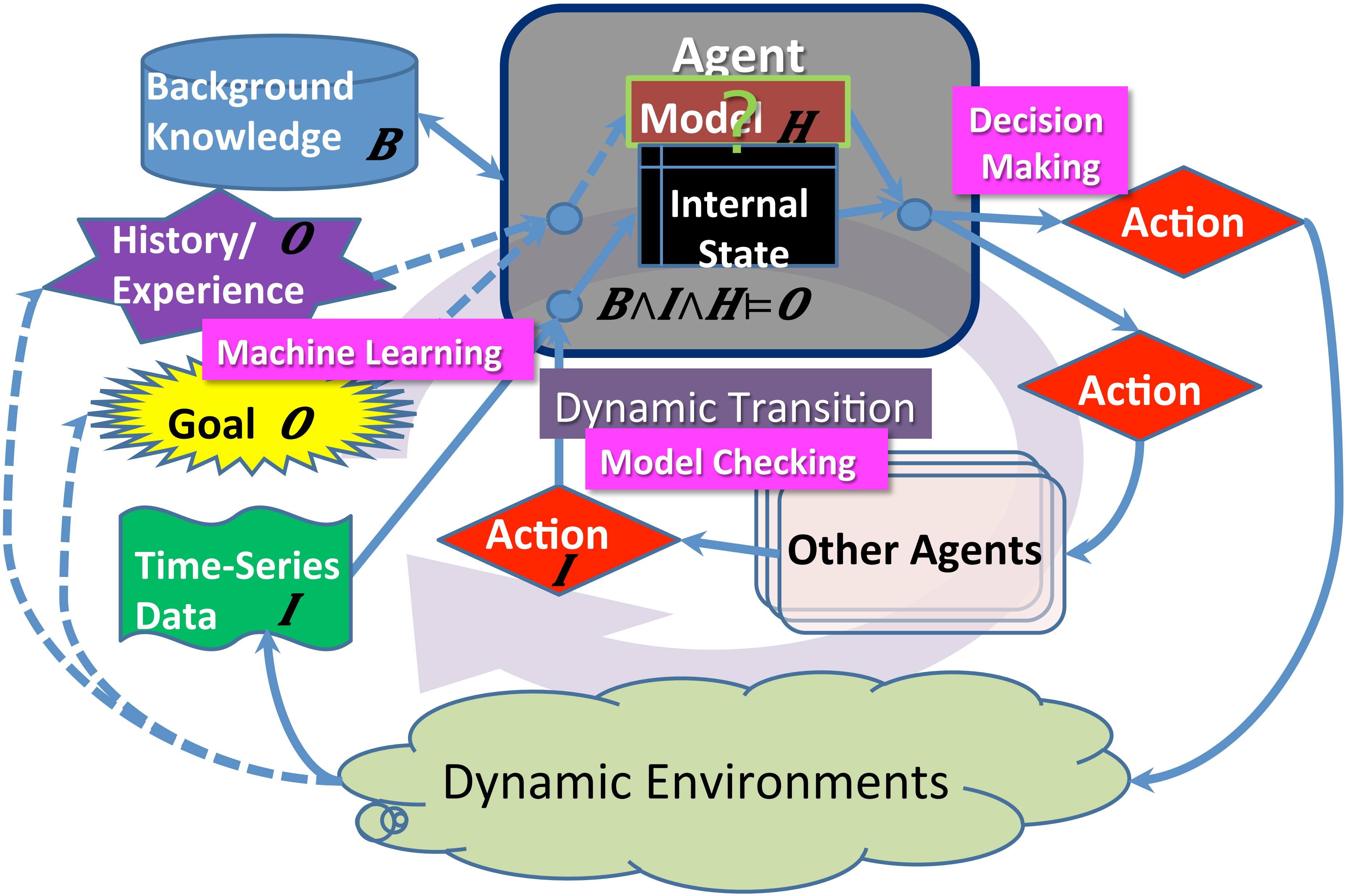


Comet, *et al.*, "Simplified models for the mammalian circadian clock", *Evry Spring school on Modelling complex biological systems in the context of genomics*, pp.85–106, 2012.



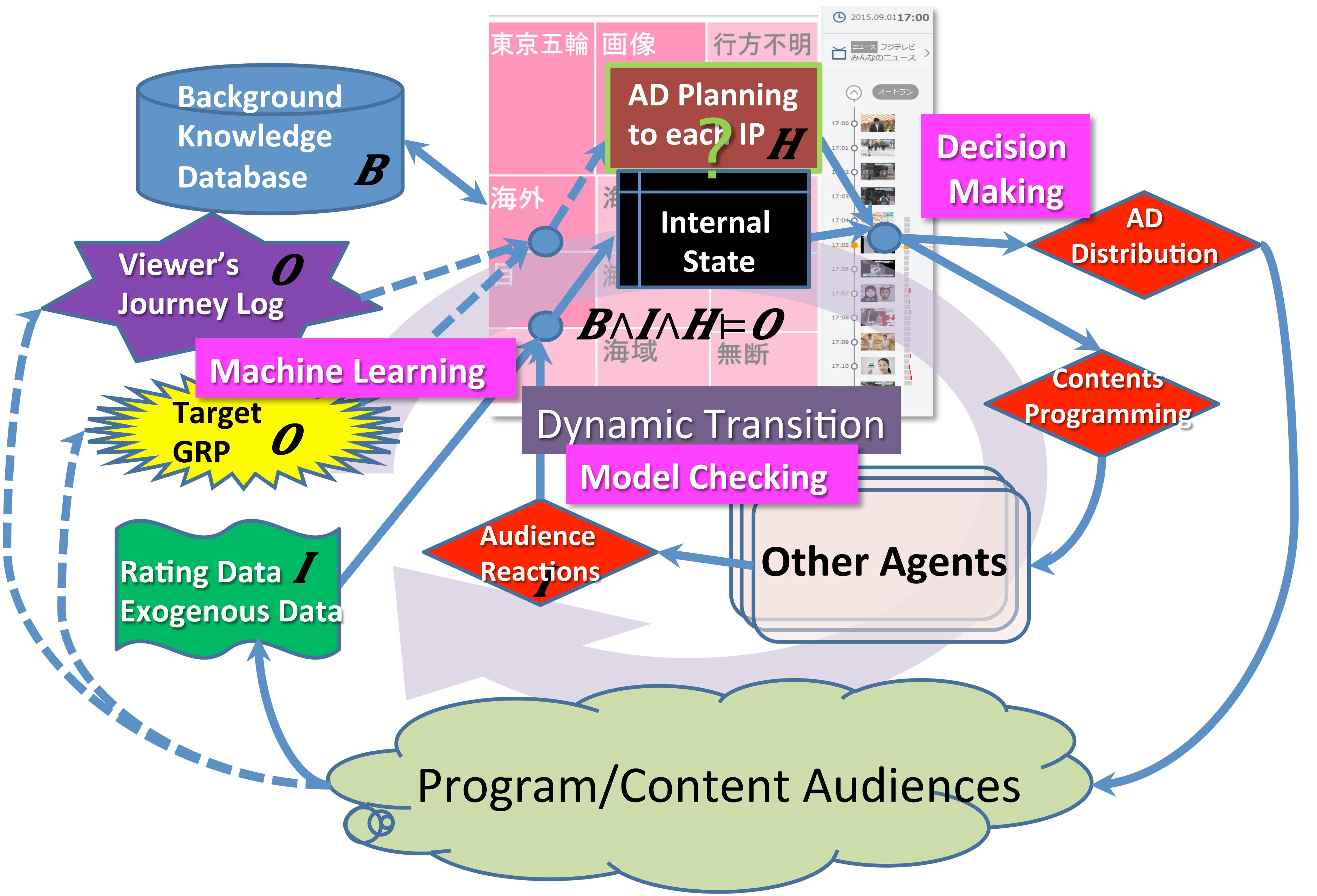
- Use *parametric time Petri nets* in order to analyze precisely the dynamic behavior of biological oscillatory systems, e.g., the mammalian circadian clock.
- Analyze resilience properties that endorse major changes in their environment, e.g. jet-lags, day-night alternating work-time.

➢ Andreychenko, A., Magnin, M., Inoue, K.: "Modeling of Resilience Properties in Oscillatory Biological Systems Using Parametric Time Petri Nets", *CMSB 2015*: 239-250.



Agent
Model ? H
Internal State
$B \wedge I \wedge H \models O$
Background Knowledge B
History/ Experience O
Machine Learning
Goal O
Time-Series Data I
Decision Making
Action
Action
Dynamic Transition
Model Checking
Action I
Other Agents
Dynamic Environments

Adaptation to IP Broadcasting 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