Probabilistic Constraint Nets

A Unified Framework for Probabilistic Hybrid Systems

Robert St-Aubin and Alan K. Mackworth, Department of Computer Science, University of British Columbia (ROPAR Project)

Motivation

The modeling and analysis of hybrid dynamical systems (a combination of analog and digital components) is increasing in importance as these models are now widely used to reason about complex physical systems.

Traditional approaches to real-time hybrid systems usually define behaviours purely as deterministic. However, real-time dynamical systems often behave probabilistically and thus exhibit uncertainty. It is therefore important to be able to model and analyze these probabilistic systems.

Our Proposal

We propose a formal framework, called *Probabilistic Constraint Nets* (PCN), for modeling, specifying and verifying hybrid systems exhibiting uncertainty.

We define a PCN as a tuple $<\mathcal{L}, \mathcal{T}, \mathcal{P}, \mathcal{C}>$ where:

- $\mathcal{L}$ set of Locations (variables) R
- $\mathcal{T}$ set of Deterministic Transductions (functions) +
- $\mathcal{P}$ set of Probabilistic Transductions
- $\mathcal{C}$ set of Connections between Locations and Transductions →

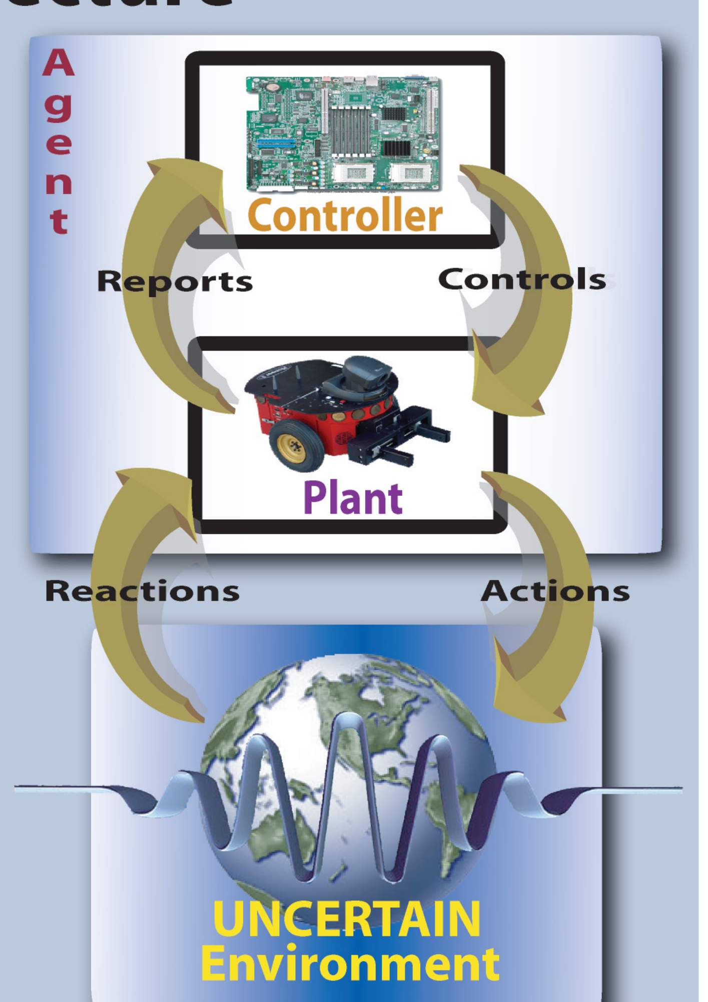
System Architecture

We view a robotic system as a probabilistic hybrid system consisting of three main PCN subsystems:

1. discrete controller
2. continuous plant

} Agent

3. continuous uncertain environment



The PCN framework is a unitary model for hybrid dynamic systems whose behaviour can be unpredictable. It includes four major components that allow for a thorough analysis of probabilistic hybrid systems.

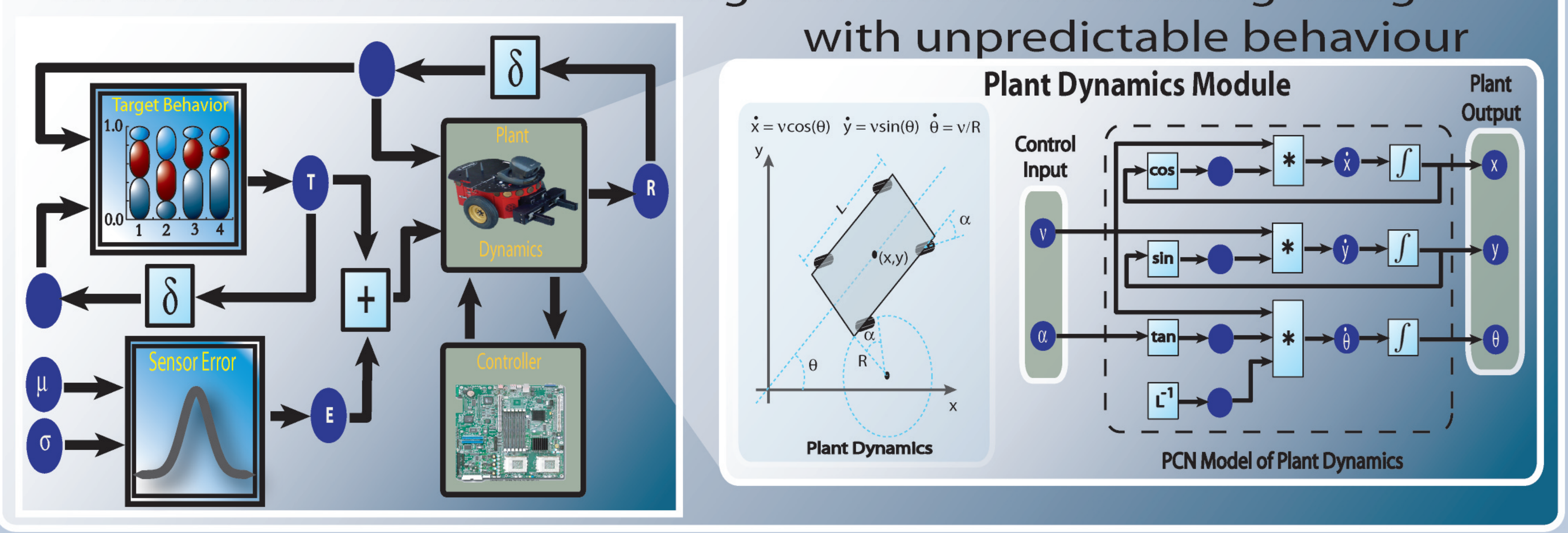
Modeling

The PCN framework greatly eases the modeling task by being: graphical modular hierarchical

PCNs generalize popular models such as: MDPs POMDPs Bayesian Nets

It allows the modeling of behaviours that are: deterministic non-deterministic probabilistic

Cat and Mouse Game: Modeling a mobile robot chasing a target with unpredictable behaviour



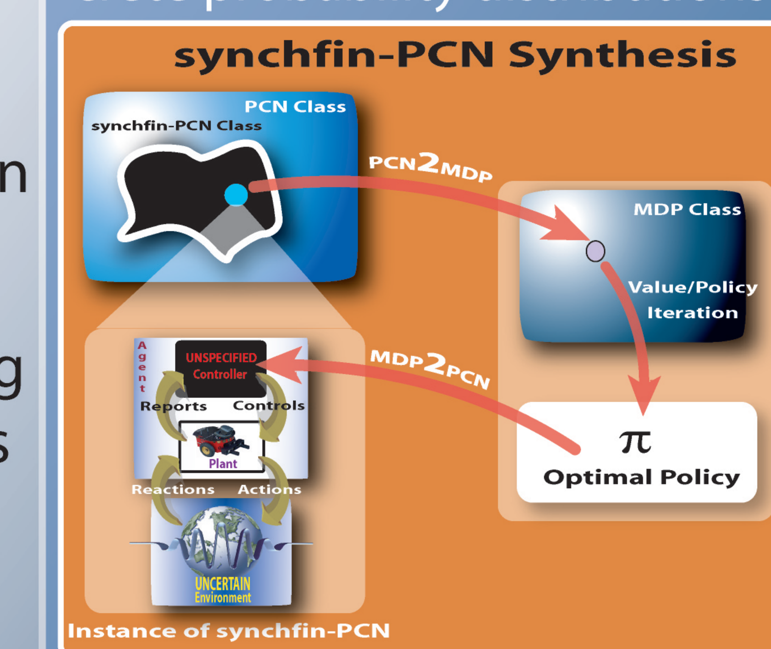
Control Synthesis

Given requirement specifications for the behaviour of a system, and models of the environment and plant, *generate* a controller that induces a behaviour satisfying the requirement.

This is difficult, especially for *probabilistic* systems, but can greatly reduce the design time.

Using dynamic programming techniques, control synthesis can be performed on a subclass of PCN: **synchfin-PCN**

synchfin-PCN: class of discrete synchronous systems with $\mathcal{P}$ inducing discrete probability distributions

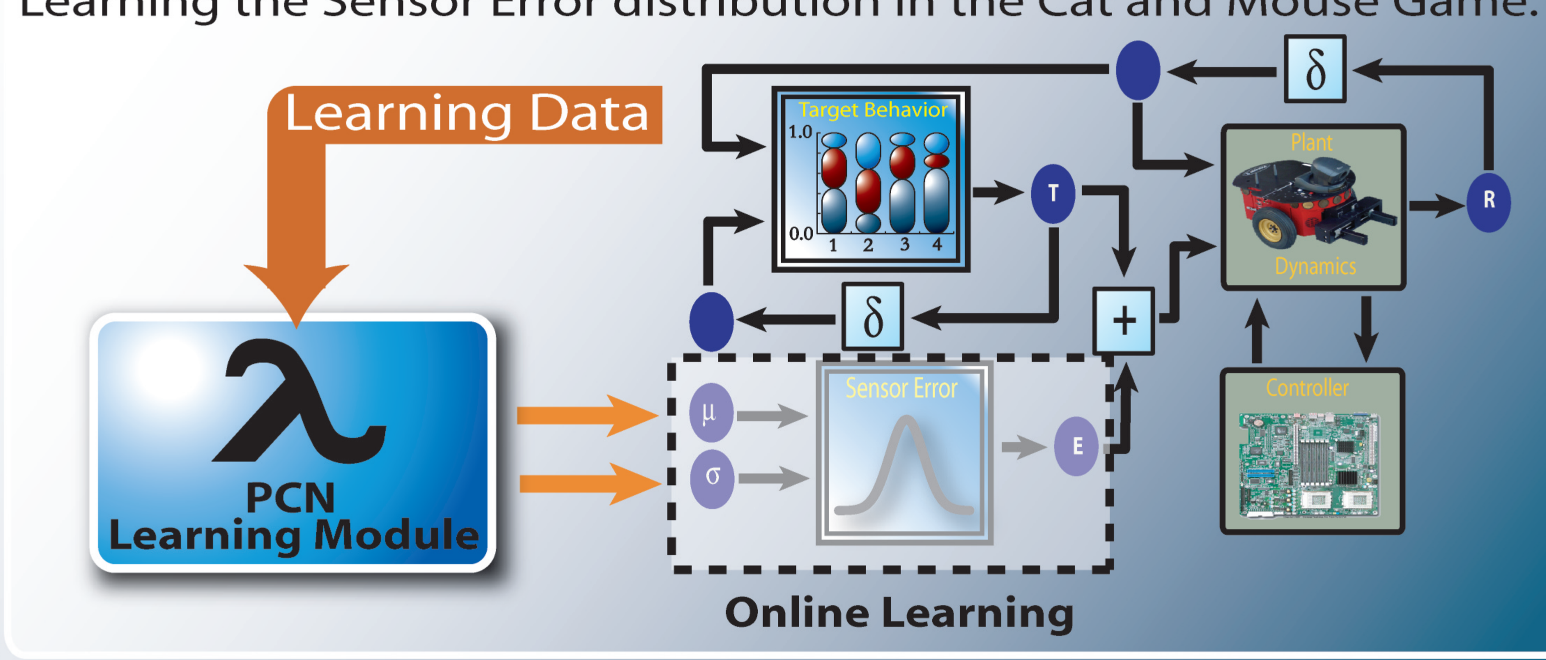


Learning

For complex systems, the optimal choice of parameters can be a very daunting task.

The PCN Learning module allows for online learning, hence automating and facilitating the parameter selection task.

Learning the Sensor Error distribution in the Cat and Mouse Game.

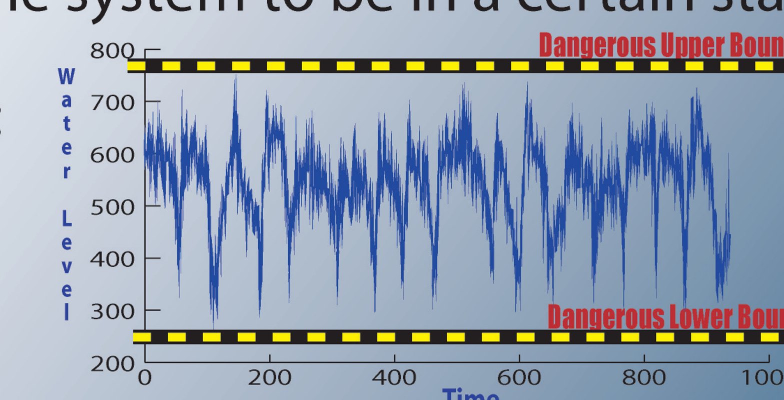


Verification

Most systems have global properties, such as *safety*, *reachability* and *liveness*, that should continuously hold.

Requirement specifications impose constraints on a system by requiring (forbidding) the system to be in a certain state.

For a steam boiler system: *"the water level should remain within* **dangerous** *thresholds"*



For probabilistic systems, it is important to determine the *probability* with which a property is satisfied. E.g.,

Conclusion

We propose a new framework, *Probabilistic Constraint Nets,* that allows one to model and analyze probabilistic hybrid systems.

The *graphical representation*, along with the *modular* and *hierarchichal* capabilities of the syntax greatly simplify the modeling task of probabilistic hybrid systems.

The framework also provides tools for performing *control synthesis*, *online learning* and *probabilistic verification*.