A Logical Framework for Self-Optimizing Networked Cyber-Physical Systems

Minyoung Kim, Mark-Oliver Stehr, Carolyn Talcott
SRI International

A Logical Framework for Networked CPS

Unifying framework

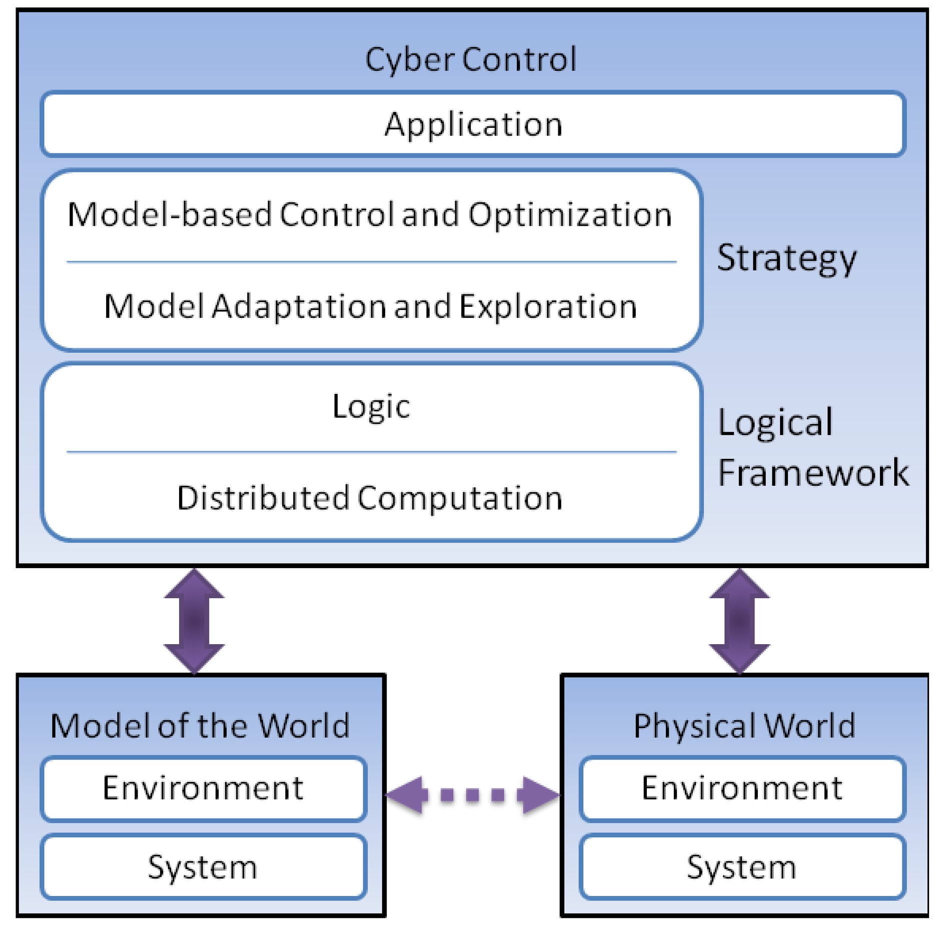
- A rigorous logical and semantic framework with degrees of satisfaction
- Unifying quantitative and qualitative reasoning and optimization
- Support robust distributed operation via knowledge sharing
- Accounting for incomplete information and uncertainty

Balance between autonomy and cooperation

- Distributed notion of goals, proofs, and proof robustness
- No need to rely on the existence or connectivity of other nodes

Distributed cross-layer strategies

- Sufficiently good solutions with acceptable resource consumption
- Sharing knowledge about solutions is essential



Distributed Cross-layer Optimization

Robustness and composability via abstraction

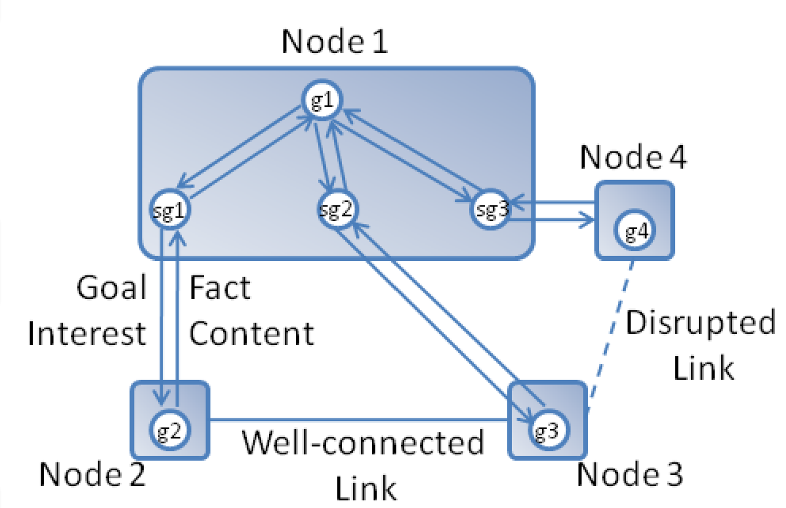
- Abstract representation of solution regions via symbolic constraints
- Inherent parallelism and fault tolerance by coordinated actions of individual nodes
- Near-optimal and robust solutions by distributed cooperative constraint refinement

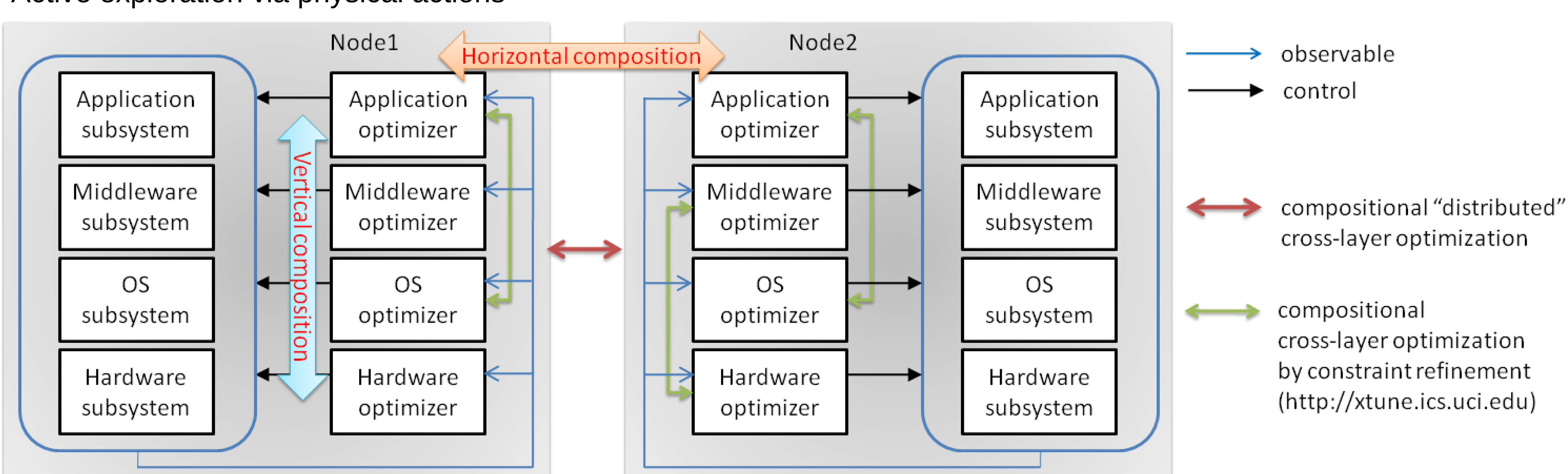
Distributed goal refinement and proof construction by strategy

- Establish intermediate goals if goal cannot be reached by a single action
- Explore multiple solution regions by combining symbolic reasoning with sampling and randomization techniques

Distributed model adaptation by observation and exploration

- Passively accumulated knowledge by observations
- Active exploration via physical actions

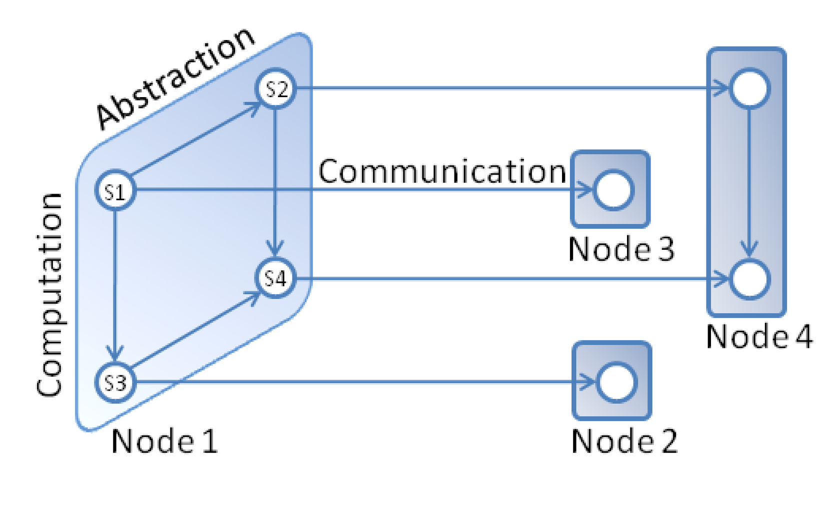




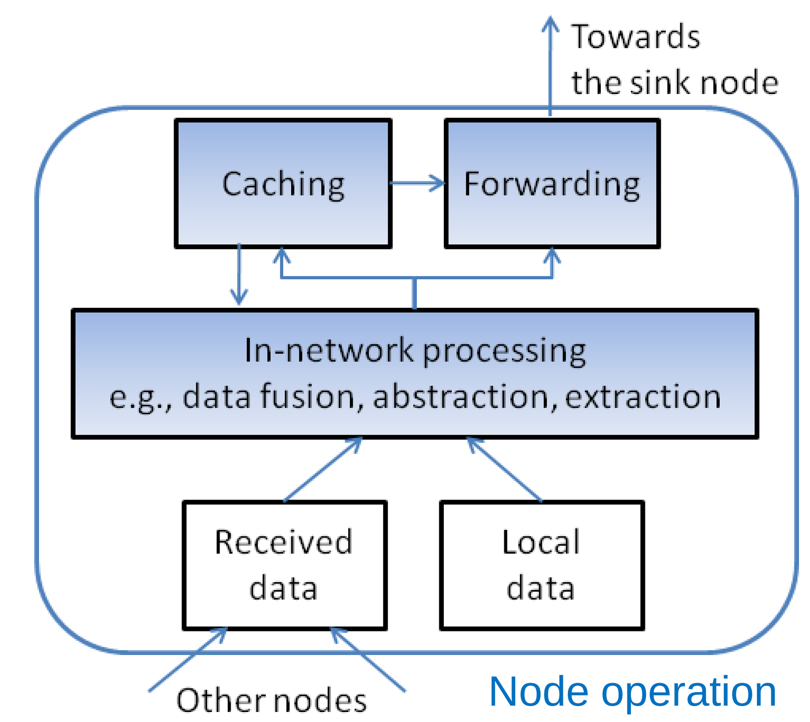
Logical Control and Optimization Strategy

Example: Interest-driven operation of Networked CPS as a logical strategy

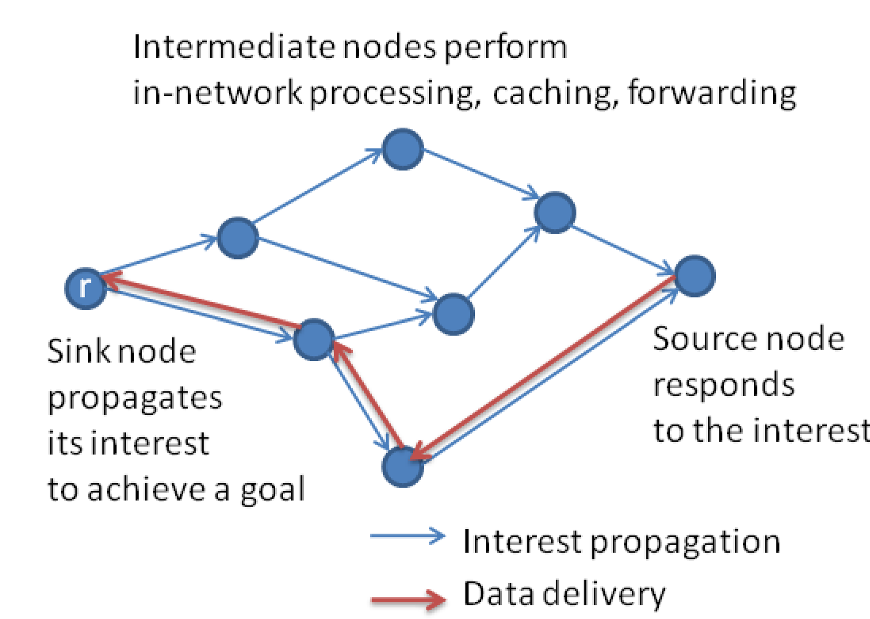
- The following goal is injected at the node *r* and disseminated through the network
  $$Motion(a,t) \vee Noise(a,t) \Rightarrow \exists I : Image(I,a,t,t+\Delta t) \wedge Delivered(Extract(Abstract(I)),r)$$
- A node in area *a* generates a fact *Motion(a, t)*, leading to the simplified goal
  $$Image(I,a,t,t+\Delta t) \wedge Delivered(Extract(Abstract(I)),r)$$
- Now, an image *i* is taken, leading to the remaining goal
  $$Delivered(Extract(Abstract(i))),r)$$
- The abstraction *Abstract(i)* → *i'* can be performed locally (but feature extraction cannot), yielding
  $$Delivered(Extract(i'),r)$$
- A more powerful node can perform the computation *Extract(i')* → *i''*, resulting in
  $$Delivered(i'',r)$$
- A goal, which can be proved by routing it toward *r* (increasing degree of satisfaction), where it is finally realized by the delivery action *Delivered(I'', r)*



Distributed computing cube

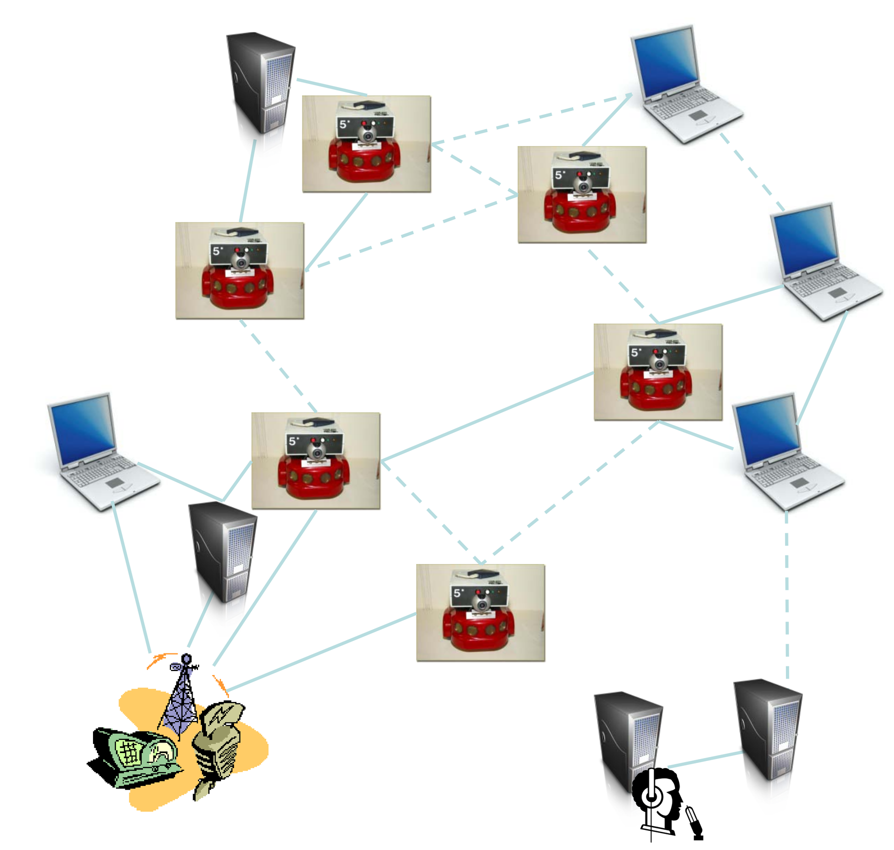


Node operation



Interest-driven routing

Test Case: Self-organizing Team of Mobile Robots



Objectives and challenges

- System may assemble and adapt on the fly for a given purpose (e.g., distributed sensing) in a resource- and situation-aware fashion
- Challenges include
  - Wide spectrum of network characteristics with mobility, delay, and disruption
  - Failure, uncertainty, partial knowledge, and stale information
  - Resource/energy constraints and tradeoffs
- Problem requires true cross-layer solutions, more general than traditional top-down (e.g., planning) and bottom-up (e.g., neighbor discovery) approaches;
  - Network topology can morph and expand driven by goals
  - Pooling resources is possible and even essential for the solution
  - Adaptive model of environment (e.g., map) is needed for intelligent control