

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

Mark-Oliver Stehr, Minyoung Kim, Carolyn Talcott

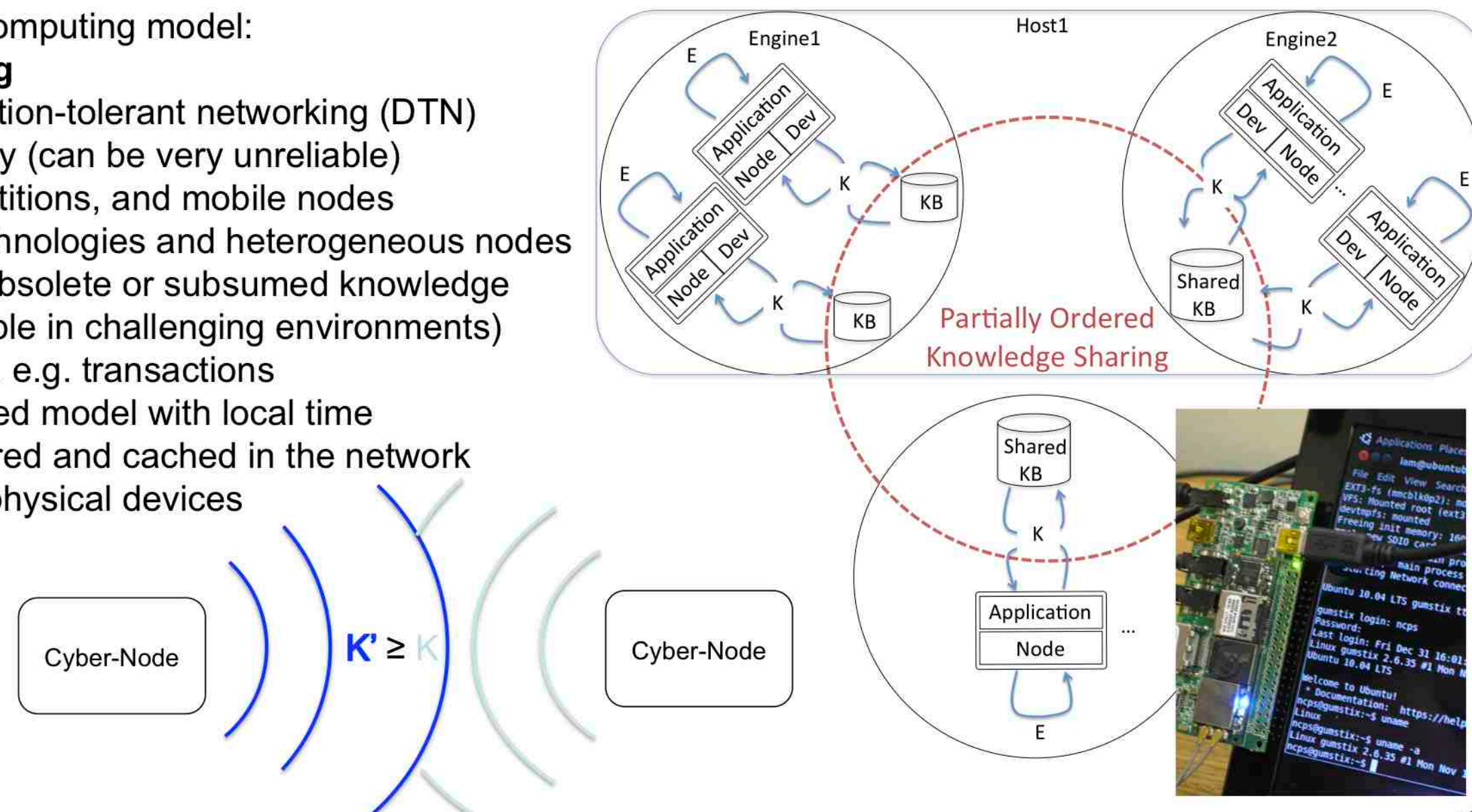
<http://ncps.csl.sri.com>

with Contributions from Jinwoo Kim (Seoul National Univ.), Vincent Wang (UPenn), Tim McCarthy (SRI)



An Application Framework for Networked CPS

- Based on new loosely-coupled distributed computing model:
 - Partially Ordered Knowledge Sharing**
- Inspired by our earlier work on delay-/disruption-tolerant networking (DTN)
- Minimal assumptions on network connectivity (can be very unreliable)
- Works with dynamic topologies, network partitions, and mobile nodes
- Designed for heterogeneous networking technologies and heterogeneous nodes
- Partial order allows the network to replace obsolete or subsumed knowledge
- Global consistency is not enforced (impossible in challenging environments)
- Avoids strong non-implementable primitives, e.g. transactions
- Locally each cyber-node uses an event-based model with local time
- Events are local, but knowledge can be shared and cached in the network
- Each cyber-node can have attached cyber-physical devices
- Framework supports
 - model-based simulation
 - probabilistic analysis algorithms
 - real-world deployment/execution
 - visualization of simulated NCPS



Networked Quadricopter Testbed

- Quadricopters are a very interesting class of cyber-physical devices (equipped with many sensors and actuators including cameras)
- Networked quadricopters will allow us to perform collaborative tasks (e.g. formation flying, distributed sensing, monitoring)
- Quadricopters (and their components) become devices in the cyber-application framework
- Currently controlled from a network of netbooks on the ground (each node can control one or multiple quadricopters)
- Can be equipped with gumstix SBC and additional devices (e.g. GPS, digital compass) for more autonomy
- Currently experimenting with vision-based localization for indoor-usage (see pictures below)



Four quadricopters before the launch



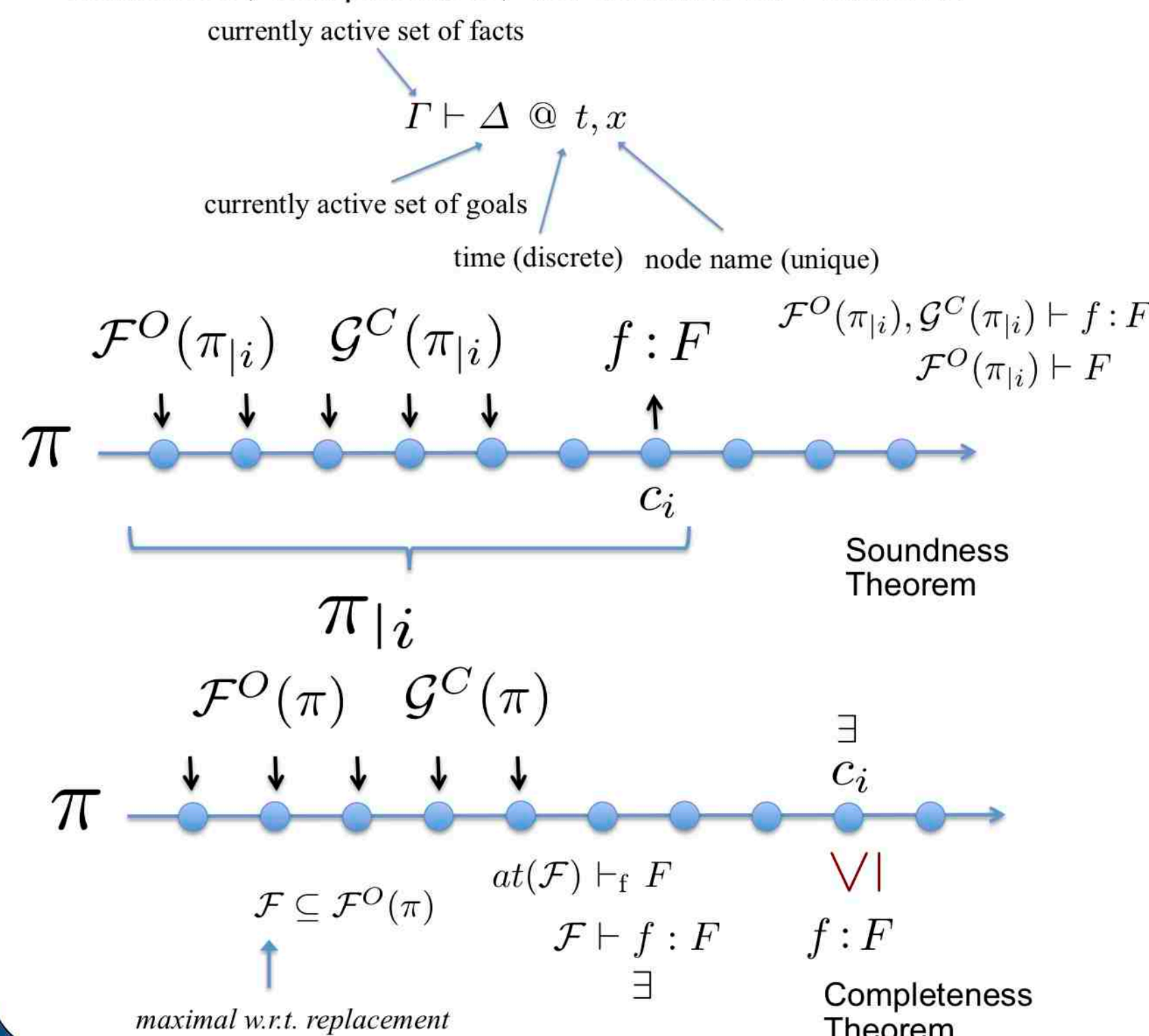
Four quadricopters controlled by the cyber-application framework



Vision-based localization experiment (utilizing Kinect 3D camera)

Distributed Logic for Declarative Control

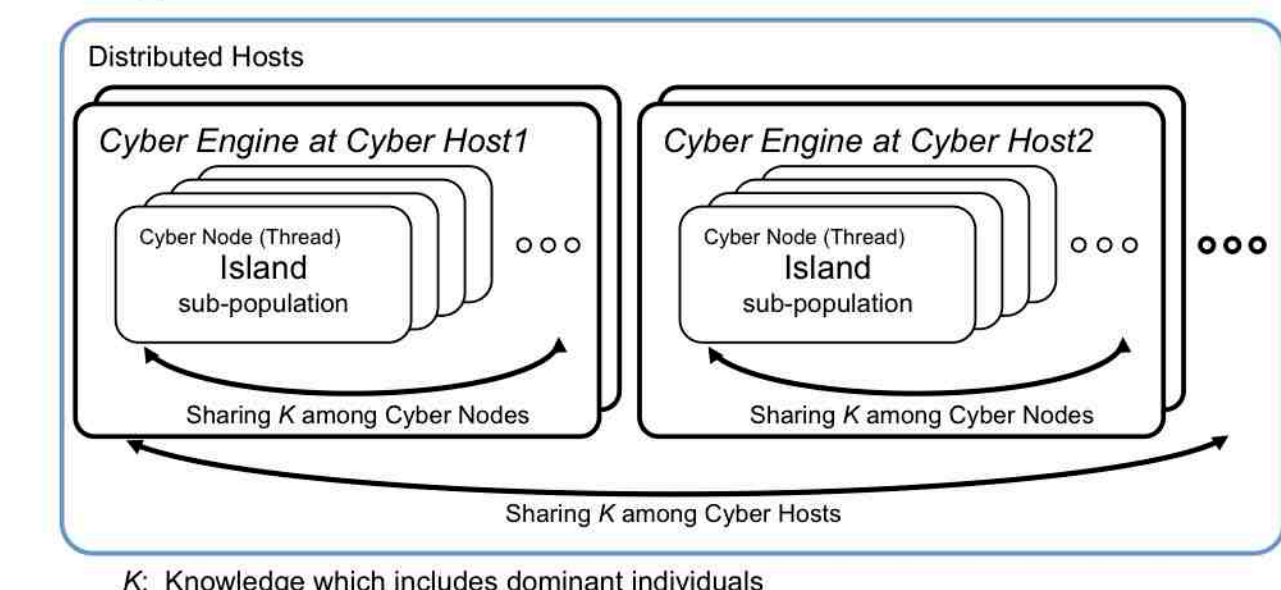
- Truly distributed logical framework with explicit proof objects
- Cyber-predicates enable interaction with the physical world
- Facts and goals treated on an equal footing
- Covers entire spectrum between autonomy and cooperation
- Tested with abstract mobility model and Stage multi-robot simulator
- Soundness, Completeness, and Termination Conditions



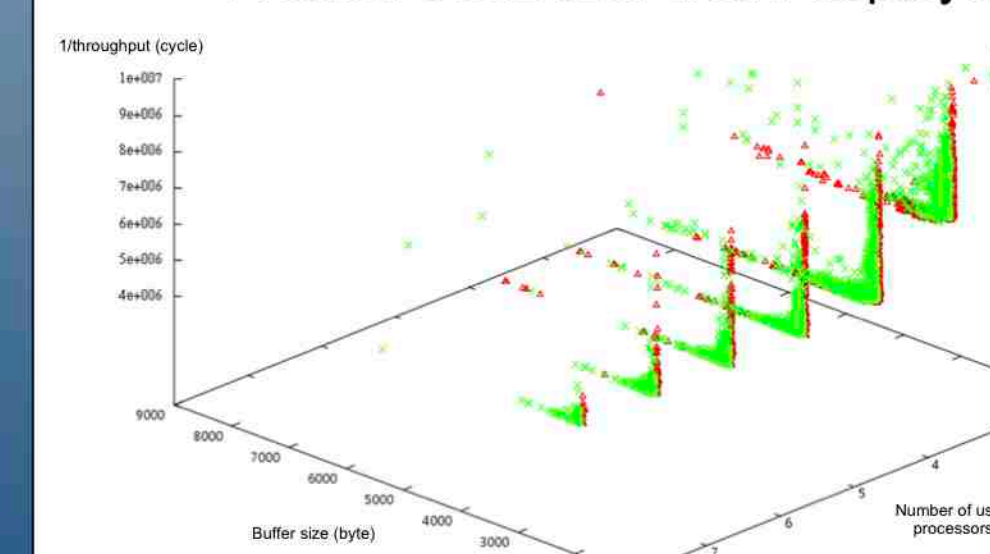
$$\begin{array}{ll}
 \frac{\Gamma \vdash \Delta @ t, x}{\Gamma \vdash \Delta, \mathcal{C}(G) : G @ t', x} & \text{if } G = p_c(t, \dots) \text{ is a cyber-goal} \quad (\text{Control}) \\
 \frac{\Gamma \vdash \Delta @ t, x}{\Gamma, 0(F) : F \vdash \Delta @ t', x} & \text{if } F = p_c(t, \dots) \text{ is a cyber-fact} \quad (\text{Observation}) \\
 \frac{\Gamma, f : F \vdash \Delta @ t, x}{\Gamma \vdash \Delta @ t', x} & \text{if } f : F \prec \Gamma, \Delta \quad (\text{Replacement1}) \\
 \frac{\Gamma \vdash \Delta, g : G @ t, x}{\Gamma \vdash \Delta @ t', x} & \text{if } g : G \prec \Gamma, \Delta \quad (\text{Replacement2}) \\
 \frac{\Gamma_x \vdash \Delta_x @ t_x, x \quad \Gamma_y, f : F \vdash \Delta_y @ t_y, y}{\Gamma_x, f : F \vdash \Delta_x @ t'_x, x} & \text{if } x \neq y, t'_x \geq t_y, \text{ and } f : F \text{ is fresh at } x. \quad (\text{Communication1}) \\
 \frac{\Gamma_x \vdash \Delta_x @ t_x, x \quad \Gamma_y \vdash \Delta_y, g : G @ t_y, y}{\Gamma_x \vdash \Delta_x, g : G @ t'_x, x} & \text{if } x \neq y, t'_x \geq t_y, \text{ and } g : G \text{ is fresh at } x. \quad (\text{Communication2}) \\
 \frac{\Gamma \vdash \Delta, g : G @ t, x}{\Gamma, B_\sigma(g) : \sigma(G) \vdash \Delta, g : G @ t', x} & \text{if } G \text{ is a built-in goal with a solution } \sigma(G) \text{ such that } B_\sigma(g) : \sigma(G) \text{ is fresh.} \quad (\text{Built-in}) \\
 \frac{\Gamma, f_1 : \sigma(P_1), \dots, f_n : \sigma(P_n) \vdash \Delta @ t, x}{\Gamma, f_1 : \sigma(P_1), \dots, f_n : \sigma(P_n), f : \sigma(Q) \vdash \Delta @ t', x} & \text{if } l : P_1, \dots, P_n \Rightarrow Q \text{ is a clause from } \Omega_t, \quad (\text{Forward1}) \\
 f = l_\sigma(f_1, \dots, f_n), \sigma(Q) \text{ is a fact, and } f : \sigma(Q) \text{ is fresh.} \\
 \frac{\Gamma, f_1 : \sigma(P_1), \dots, f_{j-1} : \sigma(P_{j-1}) \vdash \Delta @ t, x}{\Gamma, f_1 : \sigma(P_1), \dots, f_{j-1} : \sigma(P_{j-1}) \vdash \Delta, g : \sigma(P_j) @ t', x} & \text{if } l : P_1, \dots, P_n \Rightarrow Q \text{ is a clause from } \Omega_t, \quad (\text{Forward2}) \\
 g = l_\sigma^{-1}(f_1, \dots, f_{j-1}), \sigma(P_j) \text{ is a goal, and } g : \sigma(P_j) \text{ is fresh.} \\
 \frac{\Gamma, f_1 : \sigma(P_1), \dots, f_n : \sigma(P_n) \vdash \Delta, g' : G' @ t, x}{\Gamma, f_1 : \sigma(P_1), \dots, f_n : \sigma(P_n), f : \sigma(Q) \vdash \Delta, g' : G' @ t', x} & \text{if } l : P_1, \dots, P_n \Rightarrow Q \text{ is a clause from } \Omega_b, \quad (\text{Backward1}) \\
 f = l_\sigma(f_1, \dots, f_n, g'), \sigma(Q) = \sigma(G'), \sigma(Q) \text{ is a fact, and } f : \sigma(Q) \text{ is fresh.} \\
 \frac{\Gamma, f_1 : \sigma(P_1), \dots, f_{j-1} : \sigma(P_{j-1}) \vdash \Delta, g' : G' @ t, x}{\Gamma, f_1 : \sigma(P_1), \dots, f_{j-1} : \sigma(P_{j-1}) \vdash \Delta, g' : G' @ t', x} & \text{if } l : P_1, \dots, P_n \Rightarrow Q \text{ is a clause from } \Omega_b, \quad (\text{Backward2}) \\
 g = l_\sigma^{-1}(f_1, \dots, f_{j-1}, g'), \sigma(Q) = \sigma(G'), \sigma(P_j) \text{ is a goal, and } g : \sigma(P_j) \text{ is fresh.} \\
 \frac{\Gamma \vdash \Delta @ t, x}{\Gamma \vdash \Delta @ t', x} & (\text{Sleep})
 \end{array}$$

Parallel and Distributed Optimization

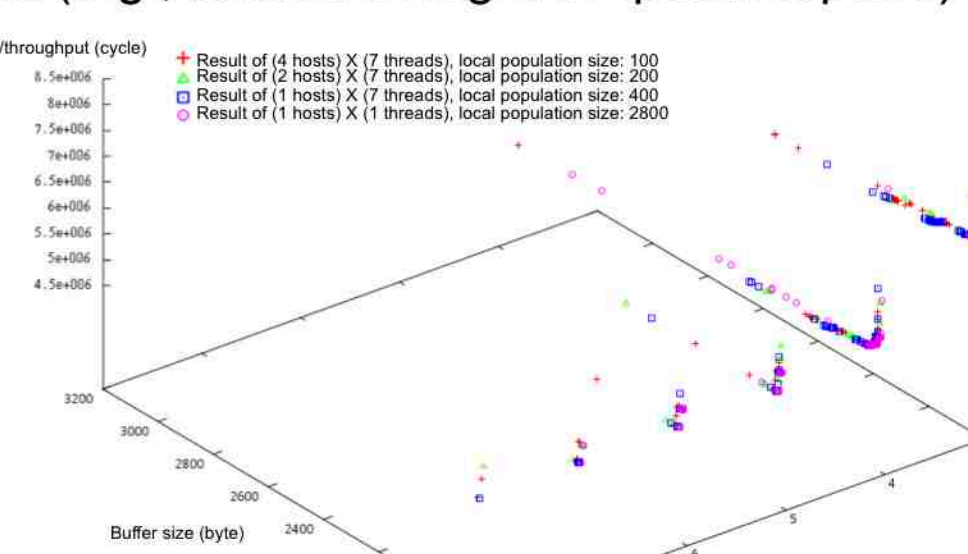
- Distributed and parallel meta-heuristic framework combining
 - an existing mature sequential optimization framework (Opt4J) with
 - a loosely coupled distributed island model for scalable parallelization
- The parallelism is transparently provided by the cyber-framework
 - cyber-nodes cooperate by emitting waves of knowledge, which interfere until all local solutions asynchronously converge to a global solution
- Optimization fits well into the partially ordered knowledge-sharing model
- Replacement order is defined by either
 - single objective function (solution fitness) or
 - multiple objective functions (Pareto optimality)
- Algorithm: population based meta-heuristic optimizer utilizing the island model
- Case study: design space exploration of an embedded multimedia system
- Key features: scalability and robustness in the optimization problem
- Optimizer performance is studied on Internet-wide testbed (Planet Lab)
- Possible next steps:
 - Combining optimization and declarative control
 - Use of weighted/quantitative/probabilistic logic
 - A small-scale real-world deployment (e.g., formation flight of quadricopters)



K: Knowledge which includes dominant individuals



Knowledge-sharing leads to better solutions within a limited time.



Solutions from parallel and sequential execution represent similar quality.

