

Optimization Congestion Control for Networked Control Systems

Ahmad T. Al-Hammouri and Vincenzo Liberatore
Case Western Reserve University, Cleveland, OH 44106 USA

Email: {ata5,vx111}@case.edu URL: <http://home.cwru.edu/~vx111/NetBots/>

INTRODUCTION & BACKGROUND

- Pervasive computing allows us to interact with remote physical environments
- Networked control systems (NCSs):
 - Using sensor data to issue control signals that affect remote physical environments
 - Distributed sensors, actuators, and controllers communicate over IP networks
 - Sensors generate a stream of sensed data and send it to controllers
 - Controllers process the samples of the sensed data and generate appropriate control signals to actuators
 - Actuators transform control signals into actions

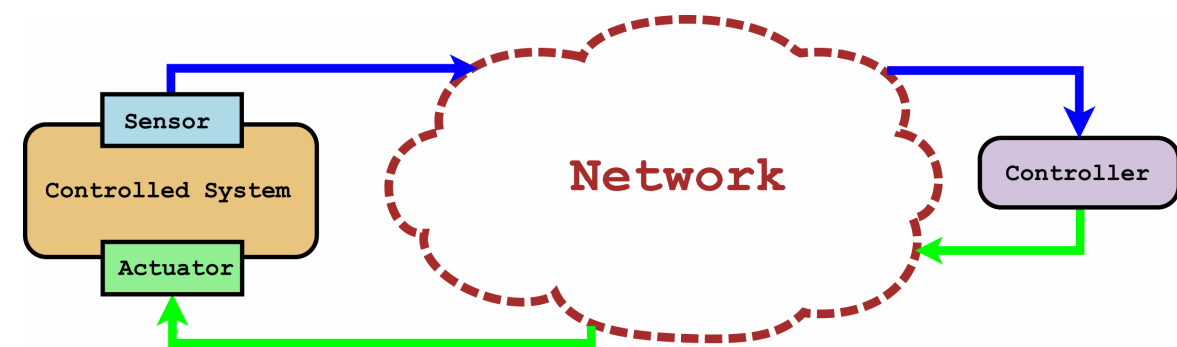


Figure 1—NCS: An example of a NCS with one system and one controller.

- Applications:
 - Surveillance in remote and hazardous areas
 - Space exploration, e.g., telerobotics
 - Medical sensing (patient monitoring) and surgical simulations
 - Industrial plants monitoring
- Advantages:
 - Increased system flexibility and reconfigurability
 - Ease of system diagnosis and maintenance
 - Reduced system wiring

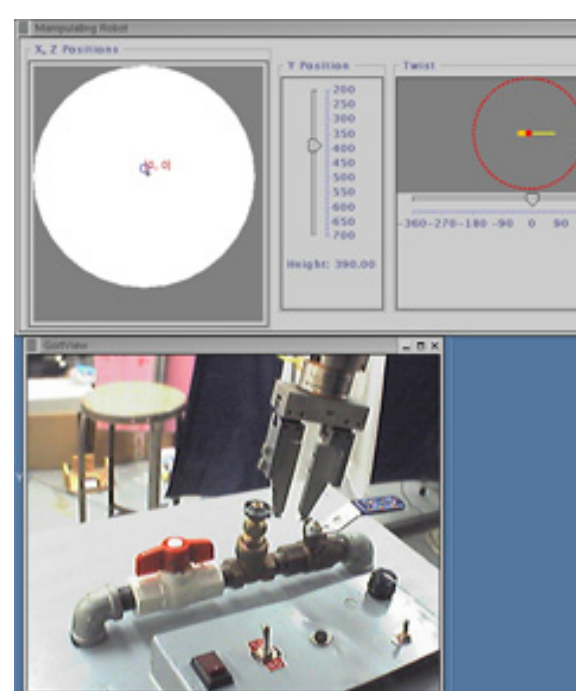


Figure 2—Internet Robotics: Teleoperating a robot through mouse movements and keyboard strokes [1]

- Challenges:
 - NCSs are real-time applications
 - Networks non-deterministic behavior and lack of QoS (e.g., delays, jitter, packet losses, and bandwidth limitation) hamper performance and effectiveness of NCS
- Approaches:
 - Control theoretical analysis, e.g., design of special controllers [6]
 - New communication network protocols, e.g., CAN

OUR APPROACH

- Multidisciplinary approach, i.e., combine study of control theory and of networking
- Objective:

Fairly allocate the network bandwidth among controlled systems to attain the maximum aggregate performance subject to stability and bandwidth constraints
- Other objectives: fully distributed solution that is scalable, flexible and reconfigurable
- Performance measure for system i : $U_i(A_i, h_i)$
 - A_i captures the physical dynamics of the system
 - h_i is the sampling period (time difference between generating two samples)
 - $1/h_i$ is the sampling rate; or the transmission rate for system i in packet/sec
 - The function U_i is monotonically decreasing with h_i
- Objective is:

$$\max \sum_i U_i(A_i, h_i) \quad \text{s.t.} \quad \sum_{i \in S(l)} \frac{1}{h_i} \leq C_l, l=1, \dots, L \quad \text{and} \quad h_{\min} \leq h_i \leq h_{\max} \quad \dots(1)$$
 where
 - $S(l)$ is set of systems whose flows have an end-to-end path that uses link l
 - C_l is the capacity of link l
 - L is the total number of links in the network
 - h_{\min} is the lower bound on h_i (which is the reciprocal of the link bandwidth connecting system i)
 - h_{\max} is the upper bound on h_i such that the system remains stable (depends on physical dynamics of system i [3])

REM: ACTIVE QUEUE MANAGEMENT

- Implementation of a distributed solution for a nonlinear optimization problem similar to Eq. (1)
- Objective is to *match* source rates to network capacity [5]
 - The queue measures congestion on the link, and feeds it back to sources
 - Sources use feedback to adjust their sending rate
- In our case, the controlled systems will use this congestion measure, *price*, to adapt their sampling rate, h_i

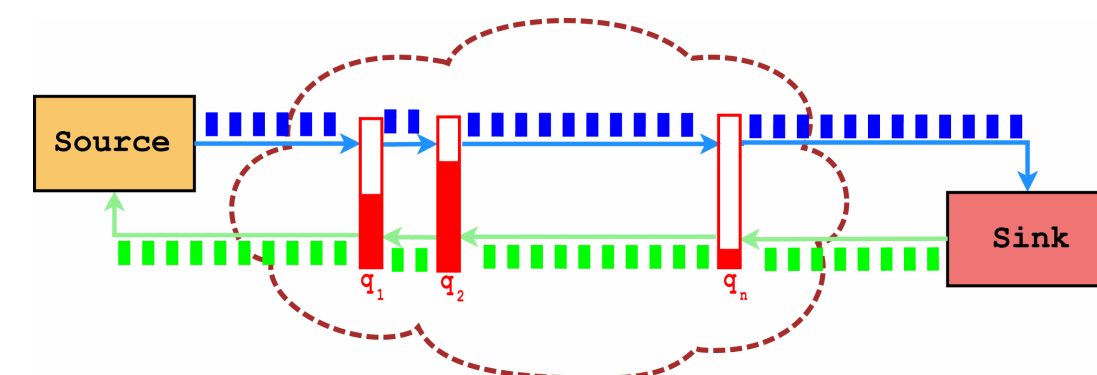


Figure 3—REM: Based on the Utility function, each source adjusts its sending rate using the price (congestion measure) being fed back from queues along the path from source to destination.

CASE STUDY: SCALAR LTI SYSTEMS

- Due to their great generality and relative simplicity
- System dynamics evolve as: $\frac{dx}{dt} = a \cdot x(t) + u(t) \quad \dots (2)$
 - where
 - $x(t)$ is the system state
 - a is the system constant— a is larger for faster system dynamics
 - $u(t) = -kx(t_i)$ is the control signal, which is simply a constant, k , multiplied by the last sampled signal
 - Necessary stability constraint: $k_i > a_i$
- A representative performance function: $U(A_i, h_i) = \frac{a_i - k_i}{a_i} e^{a_i h_i} \quad \dots (3)$
 - Based on the error in response to a unit step input that the system develops when sampling at rate $1/h_i$
- Sufficient stability constraint (if no delays) [3]: $h_{\max} = \frac{1}{a} \ln \left(\frac{k+a}{k-a} \right) \quad \dots (4)$

SIMULATION SOFTWARE, EXPERIMENT & RESULTS

- Agent/Plant**, an extension to ns-2 [4]:
 - The interface between the physical and the network dynamics
 - Can take the role of a sensor, controller interface, or actuator
- Experiment**: three plants (controlled systems) and three controllers
 - Topology: Fig. 4; Plants' Specifications: Table 1; and Results: Figs. 5 and 6

	Plant ₁	Plant ₂	Plant ₃
a_i	0.01	0.005	0.0003
k_i	1.5	0.8	0.4
Start time	0	80	40
End time	120	200	160

Table 1—Plants' Specs.: Physical dynamic parameters, and sampling start and end times for the three plants used in the experiment.

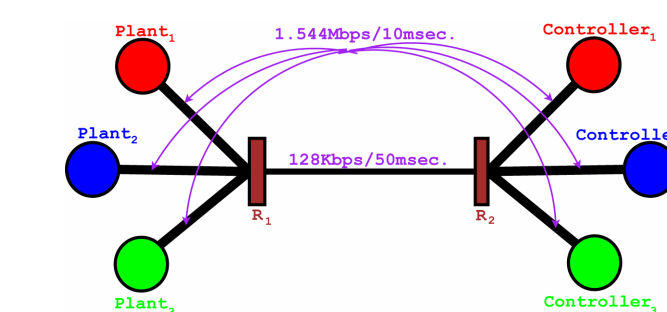


Figure 4—Simulation Topology: Agent/Plant is used to simulate three plants and three controllers sharing a bottleneck link.

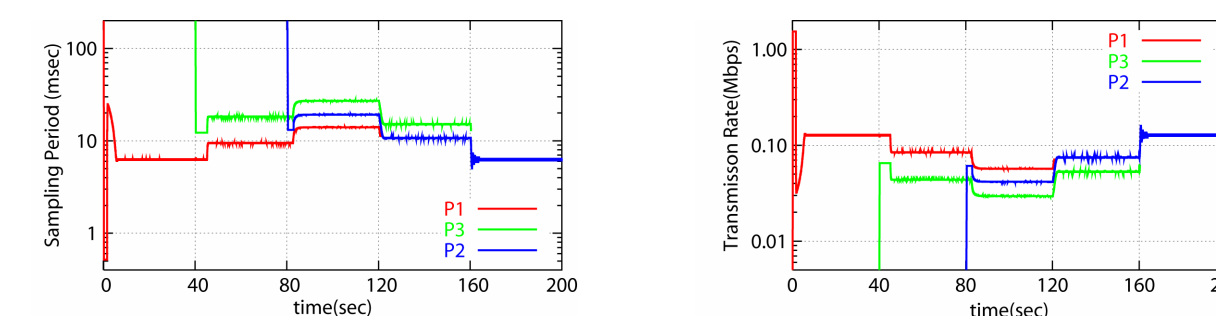


Figure 5—Sampling Period (left) & Sampling Rate (right): Each plant adapts its sampling period (left), and thus its sampling rate (right) as other plants enter and leave the network. Plants share the network based on each's performance function.

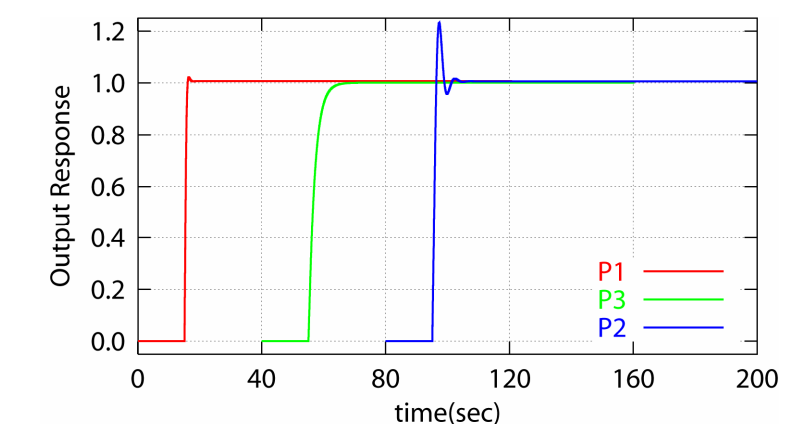


Figure 6—System Response: Plants' output in response to unit step inputs. All the plants remain stable and converge to the steady state value.

FUTURE DIRECTIONS

- Try different ways to compute *price* in REM
- Extend experiments to multi-dimensional linear plants
- Study the effect of Internet cross traffic (e.g., [2])
- Try other performance functions
- Implement these ideas in a real test bed environment

ACKNOWLEDGMENTS

- Work supported in part under NSF grant number CCR-0329910 and Department of Commerce grant number TOP 39-60-04003. Travel for presentation supported in part through financial support from School of Graduate Studies at Case Western Reserve University
- We thank Michael S. Branicky and Stephen M. Phillips for helpful conversations

REFERENCES

1. A. Al-Hammouri, A. Covitch, D. Rosas, M. Kose, W. S. Newman, and V. Liberatore. Compliant Control and Software Agents for Internet Robotics. In *Eighth IEEE International Workshop on Object-oriented Real-time Dependable Systems (WORDS)*, 2003.
2. Brian P. Robinson and V. Liberatore. On the Impact of Bursty Cross-Traffic on Distributed Real-Time Process Control. *WFCS*, 2004.
3. Michael S. Branicky, Stephen M. Phillips, and Wei Zhang. Scheduling and feedback co-design for networked control systems. In *Proc. IEEE Conf. On Decision and Control*, Las Vegas, December 2002.
4. M. S. Branicky, V. Liberatore, and S. Phillips. Networked Control System Co-Simulation for Co-Design. In *American Control Conference*, 2003.
5. S. Athuraliya, S. Low, V. Li, and Q. Yin, REM: Active queue management, *IEEE Network*, vol. 15, no. 3, pp. 48–53, 2001.
6. V. Liberatore, M. S. Branicky, S. M. Phillips, and P. Arora. (2005) Networked control systems repository. [Online]. Available: <http://home.cwru.edu/ncs/>