Limited Lookahead Control for Performance Management in Computing Systems

Sherif Abdelwahed
Institute for Software Integrated Systems
Vanderbilt University

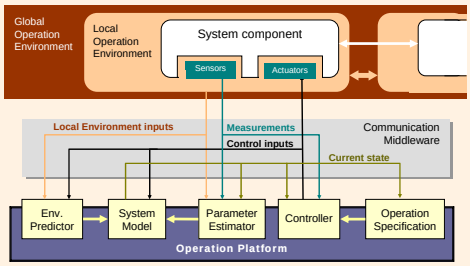
Nagarajan Kandasamy
Department of Electrical and Computer Engineering
Drexel University

The Limited Lookahead Control Framework

Motivation

- Future computing systems will typically be composed of many heterogeneous components connected via communication networks.
- Such systems will support a range of mission-critical information technology (IT) applications crucial to commerce and banking, transportation, and command and control systems.
- To cope with the complexity expected of future computing systems, it is highly desirable for such systems to manage themselves, given only high-level objectives by administrators.
- Such autonomic computing systems aim to achieve Quality of Service (QoS) objectives by adaptively tuning key operating parameters with minimal human intervention.

The Limited Lookahead Control Framework

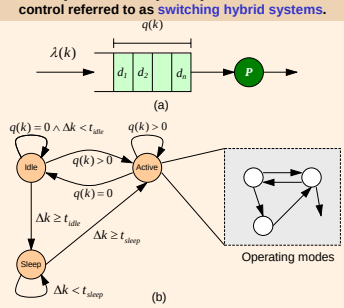


Key Aspects

- **System Model:** captures process dynamics
- **Operation Specifications:** requirements and constraints
- **Parameter Estimator:** performs online system model updating
- **Environment Predictor:** estimates future values of environment inputs
- **Controller:** Adjust system inputs to achieve desired performance

System Modeling

- Many computing systems exhibit a multi-mode hybrid behavior comprising both discrete-event and time-based dynamics.
- In addition, in most practical situations, control or tuning options must be chosen from a finite set at any given time.
- The dynamics of such systems can be captured formally as class of hybrid systems with finite control referred to as switching hybrid systems.



Computation system example: (a) A queuing model of the processor and (b) a hybrid system representation of processor operating modes

The dynamics of a switching hybrid system is represented by:

$$x(k+1) = f(x(k), u(k), \omega(k))$$

Where $x(k)$ is the state at time k, $u(k)$ is the control input chosen from a finite set U, and $\omega(k)$ is the environment input.

Environment inputs are typically estimated at time k based on previous values.

Operation requirements (specification) are represented as set points or utility functions

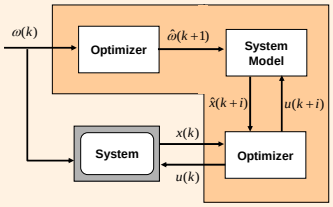
$$\hat{q}(k+1) = q(k) + \left(\hat{\lambda}(k) - \frac{\phi(k)}{\hat{c}(k)}\right).T(k)$$

$$\hat{r}(k+1) = (1 + q(k)).\frac{\hat{c}(k)}{\phi(k)}$$

$$\psi(k+1) = \phi^2(k)$$

The switching hybrid system representation of the processor at the active mode.

The Control Approach

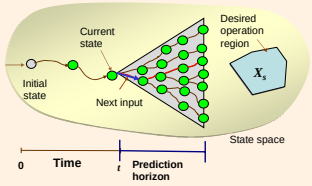


- Selection of the next step is based on a map that defines how close the current state is to X_s
- Controller constructs a tree of all future states up to certain depth.
- A path that minimizes the distance to X_s is traced back to current state and the initial step is selected.

Given a switching hybrid system with state space X and input set U. The control problem for a set point specification is to:

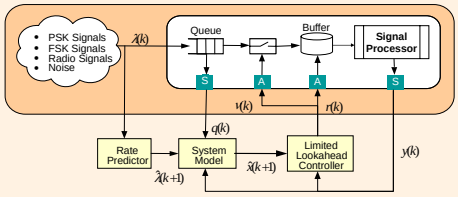
- Drive the system from any state in X to a set-point region X_s in finite time
- Maintain the system in X_s

In the case of utility function specification, the controller aims to reach the optimal state that maximizes the system utility.

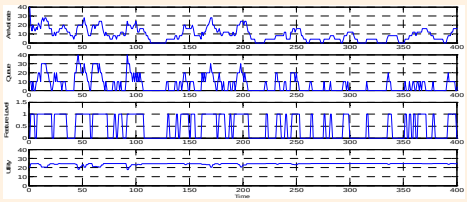


Application: Signal Detection

- **Objective:** identify relevant data from incoming signals
- Signals are received at a time-varying rate
- Detection accuracy and computation time depend on the signal size
- The controller must minimize the latency while maximizing accuracy

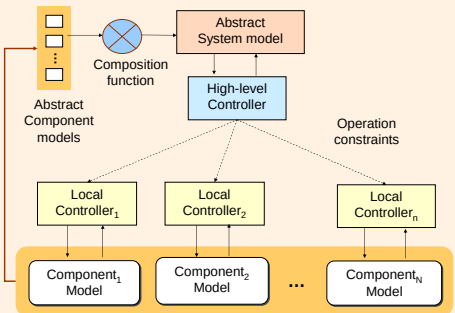


- Experiment results show that the controller adapts quickly to signal arrivals.
- In the absence of such adaptation, the queue may overflow, missing potentially interesting signals.



Hierarchical Control

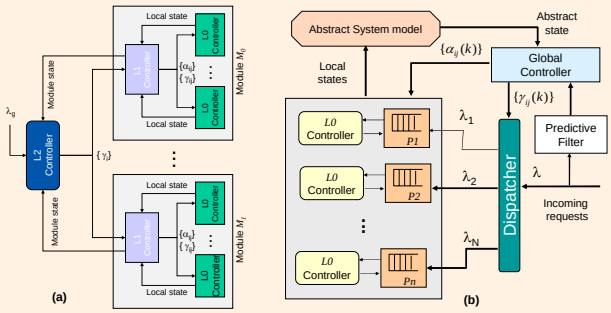
The Multilevel Control Structure



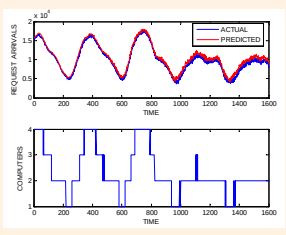
- A global controller manages inter-component interaction and enforces global requirements.
- Abstract representation of the components is used for high-level control decisions.
- Global control actions are given as additional constraints for local controllers.
- Local controllers work then to aim to optimize the performance of individual components.

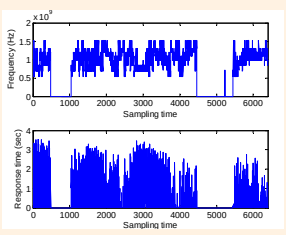
Power Management of a Computer Cluster

- **Objective:** Mange the power consumed by a heterogeneous computing cluster while satisfying QoS requirements.
- A multi-level control structure is developed where high-level limited lookahead controllers manage interactions between lower-level controllers using forecast operating and environment parameters.
- The cluster is logically partitioned into modules where each module comprises multiple processors.
- A three-level control hierarchy makes dynamic load balancing and voltage scaling decisions.



(a) The hierarchical structure imposed on a computing cluster and (b) the hierarchical structure within module M_i





- Simulations using World Cup `98 workload traces show that the architecture is scalable and adapts quickly to time-varying workload patterns.
- A module comprising four heterogeneous computers is simulated with a desired response time of 4 sec. for incoming HTTP requests.
- Multiple modules can be composed to form larger clusters.
- The average control overhead for a cluster of sixteen computers is 2.5 sec.

References

- S. Abdelwahed, S. N. Kandasamy, and S. Neema, "Online control for self-management in computing systems," 10th IEEE Real-Time and Embedded Technology and Applications Symposium, Toronto, Canada, May 2004.
- S. Abdelwahed and N. Kandasamy, "A Control-Based Approach to Autonomic Performance Management in Computing Systems," *CRC Handbook on Autonomic Computing: Concepts, Infrastructure, and Applications*, CRC Press, August 2006.
- N. Kandasamy, S. Abdelwahed, and M. Khandekar, "A Hierarchical Optimization Framework for Autonomic Performance Management of Distributed Computing Systems," *26th IEEE Int'l Conf. Distributed Computing Systems*, Lisbon, Portugal, July 2006.