

An Introduction to Networked Control Systems

Richard M. Murray (editor)
Control and Dynamical Systems
California Institute of Technology

DRAFT v1.0, March 2006
© California Institute of Technology
All rights reserved.

This manuscript is for review purposes only and may not be reproduced, in whole
or in part, without written consent from the authors.

Contents

1	Introduction	1
2	Embedded Systems Programming	7
2.1	Introduction	7
2.2	Group Communication Protocols	7
2.3	Multi-Threaded Control Systems	7
3	Trajectory Generation and Planning	9
3.1	Introduction	9
3.2	Real-Time Trajectory Generation	9
3.3	Receding Horizon Control	9
	Alice: A Networked Control Systems for Autonomous Navigation	11
.1	Introduction	11
.2	System Architecture	12
	Bibliography	17

Chapter 1

Introduction

Modern control theory is largely based on the abstraction that information (“signals”) are transmitted along perfect communication channels and that computation is either instantaneous (continuous time) or periodic (discrete time). This abstraction has served the field well for 50 years and has led to many success stories in a wide variety of applications.

Future applications of control will be much more information-rich than those of the past and will involve networked communications, distributed computing, and higher levels of logic and decision-making (see [Mur03] for a recent analysis of future directions in this area). New theory, algorithms, and demonstrations must be developed in which the basic input/output signals are data packets that may arrive at variable times, not necessarily in order, and sometimes not at all. Networks between sensors, actuation, and computation must be taken into account, and algorithms must address the tradeoff between accuracy and computation time. Progress will require significantly more interaction between information theory, computer science, and control than ever before.

An emerging architecture for networked control systems is shown in Figure 1.1. This architecture separates the traditional elements of sensing, estimation, control, and actuation for a given system across a network and also allows sharing of information between systems. As we will see in the examples below, careful decisions need to be made on how the individual components in this architecture are implemented and how the communications across the networked elements is managed. This architecture can be used to model either a single system (using either half of the diagram) or multiple systems that interact through the network.

One example of the use of this architecture is autonomous operations

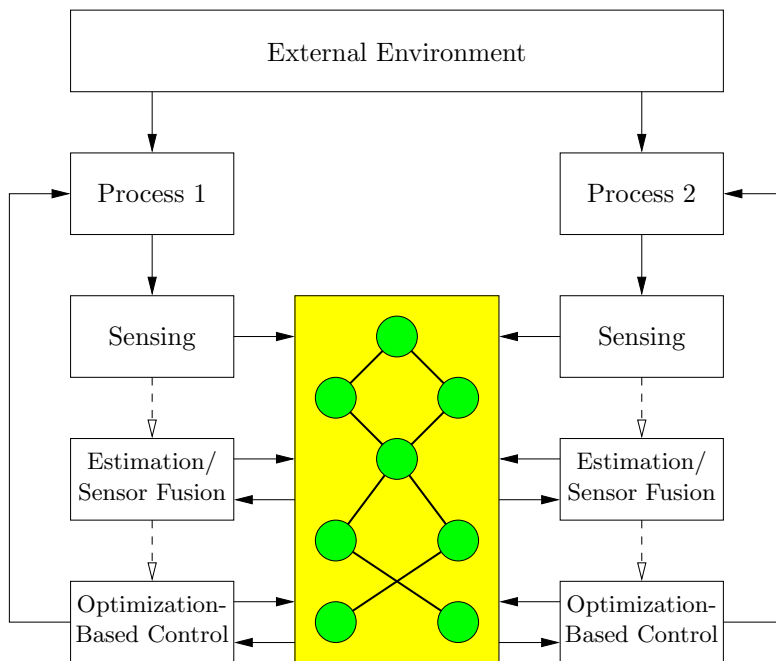


Figure 1.1: Control architecture for a networked control system.

for sensor-rich systems, such as unmanned, autonomous vehicles. As part of the 2004 and 2005 DARPA Grand Challenges, Caltech has developed two such vehicles (“Bob” and “Alice”) that each make use of a networked control systems architecture. Alice, the 2005 vehicle, has six cameras, 4 LADAR units, an inertial measurement unit (IMU), a GPS navigation system, and numerous internal temperature and vibration sensors. The raw data rate for Alice is approximately 5 Gb/s, which must be processed and acted upon at rates of up to 100 Hz in order to insure safe operation at high driving speeds.

The control system for Alice makes use of the architecture depicted in Figure 1.1, with distributed data fusion to determine elevation maps (for the height of the terrain in front of the vehicle), multiple optimization-based controllers to plan possible routes for the vehicle, and online modeling, fault management, and decision making to provide reliable and reconfigurable operation. Eight onboard computers distribute the computational load, sharing information at mixed rates across a 1 Gb/s switched network. System specifications call for reliable operation in the presence of up to 1 computer failure and 2 sensor failures, requiring careful coordination between compu-

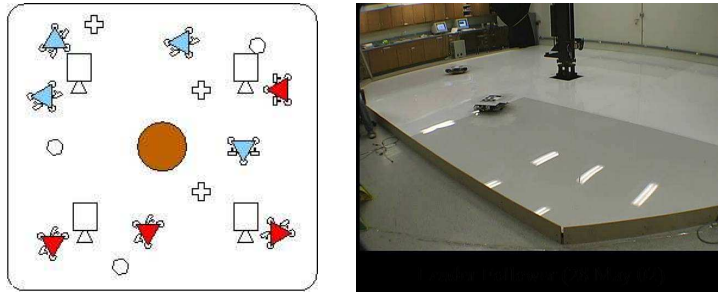


Figure 1.2: The Caltech Multi-Vehicle Wireless Testbed. The left figure shows the layout of the testbed area, including overhead cameras and fixed communication nodes (crosses and hexagons). The right picture is the current laboratory, with two vehicles shown.

tational elements.

A major challenge in Alice is determining how to send information between nodes. Because of the high data rates and computational loads on the CPUs, packets sent across the network are not always received and the system must be robust to various networking effects. Operations are currently limited to TCP, UDP and broadcast protocols, with choice of protocols determined largely by trial and error. As an example of the issues that must be resolved, certain packets of data are very important, such as packets containing raw sensor information from a portion of the terrain that is scanned only once. Other data can be dropped if needed, such as commanded trajectories (the old trajectory can be used for several sampling periods). Data from the inertial measurement unit must be received with minimum latency, while other data (a change in the temperature of the vehicle) is much less time critical. Substantial effort has been put into trying to make sure that the computations and network protocols complement each other and that loss of data and data latency does not degrade the performance of the system.

Another example of a networked control system is illustrated by the Caltech Multi-Vehicle Wireless Testbed (MVWT, shown in Figure 1.2), which consists of a collection of 8-12 vehicles performing cooperative tasks. The MVWT represents a slightly different instantiation of the architecture in Figure 1.1: each vehicle has a single processor with full access to local sensing and actuation, but information between vehicles must be sent across the network. The wireless communication channels can exhibit significant degradation when multiple vehicles are attempting to communicate and packet loss rates of 5-15% are not uncommon.

Optimization-Based Control

Mode Selection and Fault Management

Local Control

Network Communications

Online Model

Timing Management

Goal and Contingency Management

Attention and Awareness

Adaptation and Learning

