

On Design Methods for Sampled-Data Systems

Geir Dullerud and John Doyle *

Abstract

In this paper we compare, via example, the standard approaches to sampled-data design with recently developed direct design methods for these hybrid systems. Simple intuitive examples are used to show that traditional design heuristics provide no performance guarantees whatsoever. Even when the sampling rate is a design parameter that can be chosen as fast as desired, using design heuristics can lead to either severe performance degradation or extreme over-design. These effects are apparently well-known to practitioners, but may not be widely appreciated by the control community at large. The paper contains no new theoretical results and is intended to be of a tutorial nature.

1 Introduction

Sampled-data systems are hybrid systems consisting of interacting discrete-time and continuous-time subsystems, for example when digital hardware is used to implement a controller for a continuous-time process. Sampled-data systems also arise because some systems only permit discrete measurements; notably, for many chemical and biomedical systems the only sensing instruments available produce discrete signals.

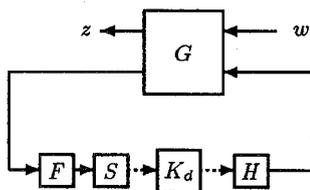


Figure 1: General Sampled-data Configuration

Figure 1 shows a widely used model for sampled-data systems: G is a continuous-time plant, which is to be controlled by the discrete-time controller K_d ; the S and H blocks in the diagram are sample and hold devices, respectively, and F is a low-pass data filter. The objective of the controller is to attenuate the gain from the input w to the output z . The sampled-data design problem is therefore:

Given a continuous-time plant G , find a discrete-time compensator K_d that achieves the performance requirements.

Sampled-data systems have been studied since the 50s (see, e.g. [11]), and there are three basic approaches to the above design problem. These are illustrated in Figure 2. The traditional approaches to sampled-data design follow the paths on the perimeter of Figure 2: one of these approaches is to design a continuous-time controller K as the first step; then to discretize K using a heuristic to arrive at the compensator K_d . The second approach is to discretize the plant G , and then to perform a discrete-time design on the plant G_d to obtain K_d . The former method ignores implementation during the design stage, and the latter technique of dis-

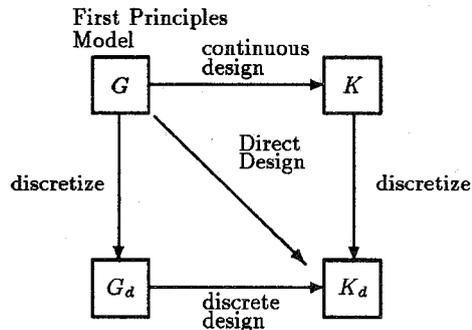


Figure 2: Design Methods

cretizing the plant does not account for intersample behavior during the design stage. The above approaches are described in the two standard texts [3] and [8], and rely on the sampling rate being significantly faster than the dynamics to be regulated. These methods have considerable practical merit in many situations, but because they are inexact there has been considerable research activity recently (see references for a partial list) into *direct sampled-data design*. This is shown by the diagonal line in Figure 2. In direct sampled-data design the compensator K_d is obtained taking into account both implementation considerations and intersample behavior.

Sampled-data design problems can be classified into two broad categories:

- the sampling rate is a fixed parameter
- the sampling rate is a design variable

In the first of these, the designer is given a fixed sampling rate for the compensator, and must produce a design that operates at this rate. The advantages of using direct sampled-data design are clear in this scenario when the sampling rate specified does not satisfy engineering rules-of-thumb or heuristics. However, when the sampling rate can be chosen by the control engineer the benefits of direct design are not (at first) so apparent.

In the next section we present two simple examples that clearly illustrate these points. A continuous-time design example demonstrates that even at high sample rates, standard design heuristics provide no performance guarantees. A discrete-time example shows that considering the open-loop bandwidth of a system can lead to extreme over-sampling in the controller implementation. The examples used are intentionally very simple and naive, so that the problems in the heuristic designs are easily recognized and understood. This also means that they are readily correctable by recognizing special structure in the problem and hand-crafting a solution. This task becomes complex, and perhaps impossible, in large multivariable designs. Direct design methods account for both implementation and intersample behavior in an automated procedure. While this may be unnecessary in many situations, it is convenient to have true sampled-data methods available, thus freeing the the control engineer to focus her engineering judgement on tasks where it is necessarily required.

2 Examples

To construct the examples we use the loop-shaping arrangement of Figure 3, where performance is measured by the L_2 induced norm of $w \mapsto z$. When T and its left-inverse

*the authors are with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125

