Noisy Timing Channels with Binary Inputs and Outputs*

Keye Martin and Ira S. Moskowitz

Center for High Assurance Computer Systems, Code 5540
Naval Research Laboratory
Washington, DC 20375
{kmartin,moskowitz}@itd.nrl.navy.mil

Abstract. We develop the algebraic theory of timed capacity for channels with binary inputs and outputs in the presence of noise, by obtaining a formula for capacity in terms of the unique solution of a nonlinear algebraic equation. We give provably correct numerical algorithms for solving this equation, specifically tailored toward calculating capacity. We use our results to establish that information theory has an inherent discontinuity in it: the function which assigns the unique capacity achieving distribution to the noise matrix of a binary channel has no continuous extension to the set of all noise matrices. Our results provide new formulae in the case of untimed binary channels as well. Our results are important in the study of real-world systems, such as the NRL Network Pump® system and traffic analysis in anonymity systems.

1 Introduction

A *timing channel* is a covert channel[1] [6] where the output symbols are distinct time values — information is passed only via the concept of time. In [14] simple timing channels (STCs) were analyzed. An STC is a noiseless covert timing channel. The fact that capacity (C) of STCs can be calculated from both a mutual information—expected time (asymptotic) and an algebraic approach was of course first done by Shannon and further analyzed in [14]. Up until this paper, with one exception, the capacity of *noisy* timing channels could only be studied via the mutual information—expected time approach. The exception to this was the analysis of the timed Z-channel in [12]. In [12] it was shown that the capacity of a timed Z-channel, which is the simplest version of a timing channel with noise, could be given as the base two logarithm of the root of a polynomial. Thus mimicking Shannon's results for STCs. This ability to view the capacity in an algebraic sense is extremely appealing.

Once an algebraic formulation of capacity is obtained, one has at their disposal a tool that can be used to study and learn about all channels, as opposed

* Research supported by the Naval Research Laboratory.

[1] Unless noted otherwise all channels in this paper are both discrete and memoryless (DMC) (which also implies stationary distributions).

J. Camenisch et al. (Eds.): IH 2006, LNCS 4437, pp. 124–144, 2007.

to just a particular channel. For instance, in [14], the algebraic approach makes many capacity relations obvious that would normally be obfuscated by viewing the capacity as simply the maximum of the ratio of the mutual information (in bits per symbol) to the expected time (units of symbols per unit time). Another advantage to the algebraic formulation is that it allows one to develop algorithms for calculating timed capacity and to prove they are correct before implemented. Currently, practitioners who determine the threat posed by covert timing channels within high assurance devices do not usually perform capacity calculations because of the mathematical complications involved. So while our knowledge of timed capacity continues to advance, it is simply theory that practitioners do not benefit from. But since the algebraic approach yields provably correct algorithms for calculating timed capacity, we can develop software that will perform these calculations for them, greatly improving the current methods used to analyze covert channels in high assurance devices.

This paper shows how to algebraically derive the capacity of a noisy timing channel with two input symbols and two output symbols. We call these $(2, 2)$ *timing channels,* with the idea that they are in general noisy being implicitly understood. This is an important result in the hopeful path to attempting to show that Shannon's capacity results in general have an algebraic solution. Aside from the interesting mathematical flavor of our results, we motivate the study of (2,2) timing channels with two open problems from the high assurance computing literature. The first example is from the NRL Network Pump® system, and the second example is from the area of anonymous communications.

2 The NRL Pump

In [5] the Network Pump was discussed as a solution to a secure, reliable, pragmatic, and robust method of sending messages up from several "Lows" to several "Highs". When a Low sends to a High, message acknowledgments, or ACKs, are required for reliability. Unfortunately ACKs can be used to send information from High to Low, which is against our wishes (Low can talk to High, but High should not be able to talk to Low if we desire security). Even if the ACKs are stripped down, the timing of the ACKs forms the basis of a covert timing channel from a High to a Low. The Network Pump moderates the timing of the ACKs to moderate (but not eliminate entirely) the covert channel threat, while at the same time not degrading system performance in an intolerable manner. In this paper we will only concern ourselves with one Low and one High, the non-network version of the Pump [4]. This serves us well, because both the network and non-network versions of the Pump use the same basic algorithm. We will use the term Pump from now on.

The Pump works as follows: Low sends a message to the Pump, the Pump stores that message in the Pump buffer, the Pump sends an ACK to Low, when Low receives the ACK it sends its next message to the Pump (handshake protocol). High removes a message from the Pump buffer and sends an ACK (different from the ACK to Low) to the Pump. High also uses a handshake protocol to send