ROBUSTNESS ANALYSIS OF THE MATCHED FILTER DETECTOR THROUGH UTILIZING SINUSOIDAL WAVE SAMPLING

A Thesis

by

JEROEN STEDEHOUDER

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

May 2009

Major Subject: Electrical Engineering

ROBUSTNESS ANALYSIS OF THE MATCHED FILTER DETECTOR THROUGH UTILIZING SINUSOIDAL WAVE SAMPLING

A Thesis

by

JEROEN STEDEHOUDER

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Don R. Halverson
Committee Members,	Deepa Kundur
	Daniel Ralph Lewis
	B. Don Russell
Head of Department,	Costas N. Georghiades

May 2009

Major Subject: Electrical Engineering

ABSTRACT

Robustness Analysis of the Matched Filter Detector

Through Utilizing Sinusoidal Wave Sampling. (May 2009)

Jeroen Stedehouder, B.S., Texas A&M University

Chair of Advisory Committee: Dr. Don R. Halverson

This thesis performs a quantitative study, derived from the Neyman-Pearson framework, on the robustness of the matched filter detector corrupted by zero mean, independent and identically distributed white Gaussian noise. The variance of the noise is assumed to be imperfectly known, but some knowledge about a nominal value is presumed. We utilized slope as a unit to quantify the robustness for different signal strengths, nominals, and sample sizes. Following to this, a weighting method is applied to the slope range of interest, the so called tolerable range, as to analyze the likelihood of these extreme slopes to occur. A ratio of the first and last quarter section of the tolerable range have been taken in order to obtain the likelihood ratio for the low slopes to occur. We finalized our analysis by developing a method that quantifies confidence as a measure of robustness. Both weighted and non-weighted procedures were applied over the tolerable range, where the weighted procedure puts greater emphasis on values near the nominal.

The quantitative analysis results show the detector to be non-robust and deliver poor performance for low signal-to-noise ratios. For moderate signal strengths, the detector performs rather well if the nominal and sample size are chosen wisely. The detector has great performance and robustness for high signal-to-noise ratios. This even remains true when only a few samples are taken or when the practitioner is uncertain about the nominal chosen.

To my Teachers, Family, and Friends

ACKNOWLEDGMENTS

I would like to take this opportunity to express my profound appreciation to Dr. Halverson. The knowledge he passed onto me is immeasurable. His complete dedication and unwavering support throughout the whole process of my research has helped made this thesis successful. I would also like to extend my gratitude to the committee members Dr. Kundur, Dr. Lewis, and Dr. Russell. Last but not least, this paper would not have been possible without the help and steadfastness of my family and friends. I am extremely grateful for having such a strong, supportive network of individuals who have offered me inspiration, positive words, and motivation. Special thanks go to my wonderful parents Kees and Trudi, and my sister Wendy for guiding and helping me reach my highest potential.

TABLE OF CONTENTS

LIST OF TABLES

TABLE | Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

Robustness is a central issue in all of today's complex systems. It plays a fundamental role in the area of signal processing. In this field, certain prior assumptions of the input characteristics are made for creating the optimal scheme to recover a signal embedded in noise. However, as we cannot predict Mother Nature, it would be unrealistic to assume the prior assumptions of these input characteristics are always predicted correctly. Therefore, having inexact prior knowledge could lead to drastic degradation in performance of such alleged optimum schemes. It is this basic motivation that provides us with the incentive to search for robust signal processing techniques; techniques that deliver good performance when assuming nominal conditions and deliver acceptable performance under non-nominal conditions. In [1], such robust techniques for signal processing are investigated and discussed.

A. Historical

Historically, much research has been carried out in the field of engineering robustness. An algorithm is considered to be robust if its performance is not too sensitive to the inexact statistical knowledge. The classical Huber-Strassen saddlepoint technique was a widely applied method for achieving a robust system [2, 3]. Although this technique still plays an important role in today's research, it has major limitations. Most importantly, this method is non-quantitative, which should be of particular interest to the practitioner [4]. It obtains an algorithm as a solution of a saddlepoint, which is considered to be "the" optimal solution. However, any small deviations

The journal model is *IEEE Transactions on Automatic Control.*

from this solution is simply not robust and cannot be measured as to how close this solution is to being robust. Clearly, robustness is not a monolithic inflexible quantity and should not be treated as one. Another limitation of this method is that it does not readily admit non-stationarity and dependency of the data samples.

As a result of these limitations, a new approach toward measuring robustness has been developed by Halverson, et. al (see examples [4, 5, 6, 7, 8, 9]). These new techniques are based on differential geometric methods; they are naturally quantitative and readily admit non-stationarity and dependency of the data samples. This geometric approach has as aim to measure the robustness locally to a presumed nominal value (i.e., the most expected value) about the underlying distribution, but can be extended for nonlocal measures as well. Initially, these nonlocal measures were defined by focusing on "worst case" distributions [7]. The saddle point technique takes on this similar approach as its solution is based on what is "least favorable" [7]. However, in reality we would assume the underlying distribution to be much closer to the nominal value than the "least favorable" solution. Therefore, the saddlepoint technique is considered to be an overly pessimistic approach and is not as useful for practical purposes in the field of engineering.

Early work developed by Halverson, et. al, used Euclidean models for the parameter manifold. However, recent work admitted the use of non-Euclidean models [8]. Although employing this technique complicates the mathematics, nevertheless it does allow us to better model reality and eliminate certain limits. A classical example to clarify this concept is when we have a covariance matrix where the entries are imperfectly known. One approach for coping with this problem would be to allow the entries to vary by $\pm\epsilon$. If the nominal matrix is positive definite, then for small enough changes in ϵ, the covariance matrix will still be positive definite. However, how small does ϵ have to be for this matrix to remain positive definite? Given the fact that the practitioner might have to take large variations into consideration, we can see that in such cases the Euclidean model will not always suffice. In [8], the non-Euclidean model was employed to measure the robustness for linear estimation using slope with biased perturbations and slope with unbiased perturbations. In this thesis, we interpret the robustness of the matched detector using the slope with unbiased perturbations since this allows us to freely vary the variance (θ) about some fixed nominal value (θ_0).

B. Research Objective

The matched filter is the optimal linear filter for maximizing the signal-to-noise ratio (SNR) for signals embedded in additive stochastic noise. The matched filter detector, also referred to as a matched detector, is designed based on the matched filter. The matched detector has played an important role in the field of engineering. Currently, researchers have made numerous extensions and/or modifications to the matched detector model in order to compliment their field of study (see examples [10, 11, 12]). Essentially, a matched detector correlates or matches a nominal signal with a measured signal to filter out the noise in order to detect the signal. However, despite all of its suitable applications, we believe the basic matched detector has not been analyzed up to satisfactory standards with regard to its robustness; it lacks the quantitative analysis. Thus, our goal in this thesis is to analyze the robustness of the matched detector based on the quantitative methods developed by Halverson, et. al.

C. Overview of the Thesis

This thesis is organized as follows:

Chapter II presents a brief introduction to signal detection and signal estimation theory. Our main focus is on signal detection theory as we introduce the basic fundamentals and tools to be utilized later for the robustness analysis of the matched detector.

Chapter III provides the methodology used for quantifying the robustness of the matched detector. It explains how we design the detector under nominal conditions, i.e., set the threshold, followed by deriving the performance equation under non-nominal conditions from which we can obtain the detector's robustness test (characterized by a slope value) using the quantitative method. Furthermore, an intermediate analysis of the robustness is performed for both common and extraordinary variances at the end of the chapter.

Chapter IV is continuation of Chapter III as it implements a weighting method scheme used for analyzing the likelihood of certain slopes to occur. Based on these result, a brief intermediate analytical analysis is provided at the end of the chapter.

Chapter V provides two methods for measuring robustness. The first method simply performs a ratio test of some predetermined areas under the weighted slope curves in order to measure the likelihood of extreme slopes to occur. The second method provides a means of quantifying robustness as a measure of confidence signifying the certainty a practitioner can possess for its solution to be robust given this outcome is less than the nominal. Additionally, we applied a triangular weighting scheme to this method as values closer to the nominal should be given more weight as opposed to values further away from the nominal.

Chapter VI presents a discussion and conclusion about the robustness of the matched detector based on the results calculated in the previous chapters.

A list of references is included at the end of this thesis followed by the appendixes.

CHAPTER II

BACKGROUND AND FUNDAMENTALS

This chapter will introduce the basic concepts of signal detection and signal estimation theory as they are closely related subjects. Given the fact the practitioner has some type of prior knowledge, the mathematical and statistical tools developed in this theory are utilized for the extraction and processing of the available information, thus leading us to a decision with optimum accuracy. However, our focus in this thesis is signal detection theory. Therefore, we will primarily provide the fundamentals and tools necessary for the robustness analysis of the matched filter detector with regard to this topic.

A. Signal Detection Theory

Detection theory was originally developed by radar and sonar researchers and is presently applied to areas such as pattern recognition, quality control, sonar (oil exploration), digital communications, etc. Signal detection copes with the problem of making a decision among some finite number of possible situations or "states of nature" [13]. The problem can be separated into three types of classes; known signals in additive noise, signals with unknown parameters in additive noise, and random signals in additive noise [14].

The connection between the observation and the desired information is probabilistic rather than direct [13], implying a distributional model should be used. In this thesis, we model the noise to be Gaussian and independent and identically distributed (i.i.d.). If we let X be the random variable modeled according to this Gaussian distribution, we can subsequently characterize the probability density function (PDF) of X by:

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad -\infty < x < \infty \tag{2.1}$$

where μ is the mean and $\sigma^2 \geq 0$ is the variance. In short notation, we write $X\sim N(\mu, \sigma^2)$ to represent that the random variable X is normally distributed with mean μ and variance σ^2. We can see the PDF in (2.1) gives us the "point probabilities" [15], these being equal to zero in the continuous case. However, if we are interested in obtaining the probability over a specific interval, we integrate $f_X(x)$ in (2.1) over the interval of interest in order to obtain the cumulative distribution function (CDF). For instance, if we want to find all the values of x that are less than or equal to the random variable X, we write:

$$P(X \leq x) = F_X(x) = \int_{-\infty}^{x} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(t-\mu)^2}{2\sigma^2}}\, dt \tag{2.2}$$

1. The Detection Problem

A simple problem in detection theory involves the making of a decision whether or not a signal is present, i.e., we decide whether we receive noise only or a signal corrupted by noise. An example of this problem would be the detection of an aircraft based on the signal received by the radar. We can model this situation as a *binary hypothesis testing problem*, where we need to make a decision from the available data between the following two hypothesis, which are:

$$H_0 : \text{noise only} \tag{2.3}$$

$$H_1 : \text{signal + noise} \tag{2.4}$$

We note that the available data can be collected either continuously or discretely. In this thesis, we sample our data discretely for simplicity reasons and due to the fact that discrete time systems are currently gaining popularity. For example, let's consider two univariate i.i.d. noise densities f_0 and f_1 under H_0 and H_1 respectively. In the case where n samples are taken, the joint PDF under H_0 and H_1 could be written as:

$$H_0 : f_0(x_1, \ldots, x_n) = \prod_{i=1}^{n} f_0(x_i) \tag{2.5}$$

$$H_1 : f_1(x_1, \ldots, x_n) = \prod_{i=1}^{n} f_1(x_i) \tag{2.6}$$

Additionally, if the noise is Gaussian, we can write PDFs for both H_0 and H_1, denoted by $f(x; H_0)$ and $f(x; H_1)$ as:

$$f(x; H_0) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} \tag{2.7}$$

$$f(x; H_1) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.8}$$

where we note $f(x; H_1)$ to be $f(x; H_0)$ shifted by μ.

The detector selects H_0 or H_1 based on the detection algorithm, also known as the decision rule. In general, the optimal decision rule is one that defines a threshold, denoted by T_r, to where the probability of error is minimized in choosing either H_0 or H_1. There are two types of errors associated with the detector, namely:

$$\begin{aligned} Type\ I &: \text{choosing } H_1 \text{ when } H_0 \text{ is true} \\ Type\ II &: \text{choosing } H_0 \text{ when } H_1 \text{ is true} \end{aligned} \tag{2.9}$$

The probability of making a *Type I* error is known as the *false alarm rate* and is denoted by α. The probability of making a *Type II* error is known as the *miss probability* and is denoted by $1 - \beta$, where β is the probability of choosing H_1 when H_1 is true, also known as the *detection probability* or performance of the detector. We note that as the the "distance" between the PDFs increases, or equivalently, the signal-to-noise ratio (SNR) increases, the performance of the detector improves as well. A visual representation of the PDFs under either hypothesis and associated types of errors can be found in Fig. 1.

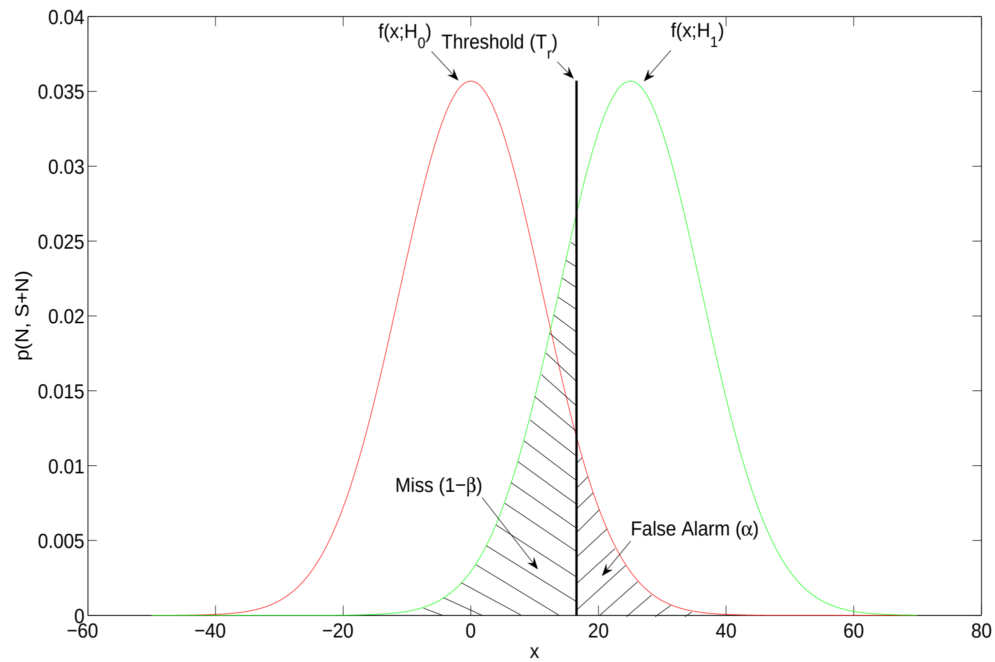

Fig. 1. Decision surface illustrated

2. Decision Rule of the Detector

There are two common frameworks used in detection theory for optimizing the decision of the detector, namely:

1. Bayesian
2. Neyman-Pearson

The Bayesian approach puts focus on minimizing the loss. For this purpose, a loss function is established to assign a loss to each possible outcome and each possible decision, where the decisions made are based on minimizing the average loss [16].

As for the parameters, they are assumed to be random variables, governed by a prior probability. On the contrary, the Neyman-Pearson (NP) framework is focused on finding a decision function which maximizes the probability of detection, given a fixed probability of false alarm. In this thesis, we restrict ourselves to the NP criterion for the robustness analysis of the matched filter detector as the NP criterion is optimal for design. In view of the NP lemma [13], we can write the decision rule of the detector to be:

$$\Lambda(\mathbf{y}) = \begin{cases} H_0, & \text{if } \lambda(\mathbf{y}) < T_r \\ H_1 \textit{ with probability } q, & \text{if } \lambda(\mathbf{y}) = T_r \\ H_1, & \text{if } \lambda(\mathbf{y}) > T_r \end{cases} \tag{2.10}$$

where $\lambda(\mathbf{y})$ is the likelihood ratio of the observation vector $y_1, \ldots, y_n$ and T_r is a deterministic constant based on the constraint α. If the joint PDF of the n samples under H_0, H_1 are $f_0(\mathbf{y})$ and $f_1(\mathbf{y})$ respectively, then we can write the likelihood ratio for n independent observations as:

$$\lambda(\mathbf{y}) = \frac{f_1(\mathbf{y})}{f_0(\mathbf{y})} = \frac{f_1(y_1, \ldots, y_n)}{f_0(y_1, \ldots, y_n)} = \prod_{i=1}^{n} \frac{f_1(y_i)}{f_0(y_i)} \tag{2.11}$$

On the other hand, if we take the natural logarithm of both the likelihood ratio and the threshold, we can obtain the *log-likelihood* ratio test which modifies the NP decision rule of (2.10) to:

$$\Lambda(\mathbf{y}) = \begin{cases} H_0, & \text{if } \hat{\lambda}(\mathbf{y}) < \hat{T}_r \\ H_1 \textit{ with probability } q, & \text{if } \hat{\lambda}(\mathbf{y}) = \hat{T}_r \\ H_1, & \text{if } \hat{\lambda}(\mathbf{y}) > \hat{T}_r \end{cases} \tag{2.12}$$

where $\hat{T}_r = \ln\, T_r$. Consequently, the *log-likelihood* ratio for n independent observations becomes:

$$\hat{\lambda}(\mathbf{y}) = \sum_{i=1}^{n} \ln\Big(\frac{f_1(y_i)}{f_0(y_i)}\Big) \tag{2.13}$$

To demonstrate the idea of a linear detector in conjunction with the log-likelihood ratio concept described above, lets suppose we have a constant signal s and i.i.d. $N(0, \sigma^2)$ noise. The distribution of the random sample Y_i under hypothesis H_0 and H_1 can then be represented by:

$$\begin{cases} Y_i = 0 + N_i = N(0, \sigma^2) \text{ under } H_0 \\ Y_i = s + N_i = N(s, \sigma^2) \text{ under } H_1 \end{cases} \tag{2.14}$$

The joint PDF of the n samples under H_0, H_1 can then be given by:

$$f_0(y_1, \ldots, y_n) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{y_i^2}{2\sigma^2}} \tag{2.15}$$

$$f_1(y_1, \ldots, y_n) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(y_i-s)^2}{2\sigma^2}} \tag{2.16}$$

Subsequently, we can define the likelihood ratio as:

$$\lambda(\mathbf{y}) = \frac{\prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(y_i-s)^2}{2\sigma^2}}}{\prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{y_i^2}{2\sigma^2}}} \tag{2.17}$$

$$= \prod_{i=1}^{n} e^{\frac{-(y_i-s)^2 + y_i^2}{2\sigma^2}} \tag{2.18}$$

and the log-likelihood ratio as:

$$\ln\, \lambda(\mathbf{y}) = \sum_{i=1}^{n} \frac{-(y_i^2 - 2y_i s + s^2) + y_i^2}{2\sigma^2} \tag{2.19}$$

Since s and σ^2 are constants, we can eliminate them from (2.19) to let the log-likelihood ratio become:

$$\ln \lambda(\mathbf{y}) = \sum_{i=1}^{n} y_i \tag{2.20}$$

At last, we are able to define the NP decision rule following to (2.12) with threshold $\hat{T}_r = (2\sigma^2 \ln T + s^2)/2s$. More information on this and others topics regarding signal detection theory can be found in literature such as [13], [14], and [17].

B. Signal Estimation Theory

Signal estimation theory was originally developed within the area of statistics and is applied to areas such as speech, radar, control systems, digital signal processing, medical science, etc. The objective of estimation theory is to estimate the values of parameters based on measured or empirical data. For example, in radar, one might be interested in estimating the location of an aircraft. To accomplish this, an electromagnetic pulse is transmitted and reflected by the aircraft, causing the radar station to receive an echo a few moments later. Clearly, we are able to measure the velocity of the electromagnetic pulse and thus, the distance between the radar station and the airplane can be computed despite the propagation losses and time delays introduced by the electronics of the receiver.

In the previous example and all other systems in general, one is faced with the problem of extracting values of parameters based on continuous-time wave forms [18]. However, we use digital computers to sample and store these continuous-time waveforms, and therefore equivalently face the problem of extracting parameter values from a discrete-time waveform or data set. From a mathematical perspective, we now have N samples stored as a N-point data set $\{x_0, x_1, \ldots, x_{N-1}\}$ which depends on an unknown parameter θ. Our goal is then to determine θ based on the data or to define an estimator such as:

$$\hat{\theta} = g(x_0, x_1, \ldots, x_{N-1}) \tag{2.21}$$

where g is some function.

There are three types of estimation [8], namely

1. Parametric estimation

2. Non-parametric estimation

3. Robust estimation

In parametric estimation, we assume that the joint and marginal distribution of the observed signal and the parameter to be estimated are known. Within this class, there are two types of estimation approaches; classical and Bayesian. In the classical approach, data is modeled by a PDF since the parameters of interest are unknown but deterministic. In the Bayesian approach, one incorporates prior knowledge of the PDF. The parameter we attempt to estimate is then viewed as a realization of the random variable θ. The data are described by the joint PDF:

$$p(\mathbf{x}; \theta) = p(\mathbf{x}|\theta)p(\theta) \tag{2.22}$$

where $p(\theta)$ is the prior PDF in where our knowledge about θ is summarized before any data are observed, and $p(\mathbf{x}|\theta)$ is the conditional PDF, where our knowledge about the data $\mathbf{x}$ is summarized given that we know θ [18].

In non-parametric estimation, one can make very few assumptions about the statistical properties of the quantities to be estimated. There is no knowledge of the exact distribution of the unknown, however, it may be possible to determine the generalized family of the symmetric distributions.

As [8] states, robust estimation has not been formally defined uniquely for a universal context. However, it is an important concept in practice since it is considered to lie in between non-parametric and parametric estimation. Basically, we do not have knowledge of the exact distribution although we must be somewhat familiar with a nominal value, i.e., a value that is most likely to occur. More information on this and others topics regarding signal estimation theory can be found in literature such as [8], [13], [17], and [18].

CHAPTER III

METHODOLOGY FOR QUANTIFYING ROBUSTNESS

This chapter presents the design process of our matched filter detector assuming nominal conditions followed by testing the robustness of our detector under non-nominal conditions. In the design phase, a threshold of the detector is derived, where in turn this result is used to specify the detection probability where from we can derive the equation that can be used to analyze the robustness of our detector.

A. Model Assumptions

Lets assume we have independent identically distributed (i.i.d.) additive white Gaussian noise $\sim N(0, \sigma^2)$. Additionally, for the detector design and robustness testing purposes, lets also assume:

1. A constant signal-to-noise ratio (SNR) K defined by:

$$K = \frac{A^2}{\theta} \tag{3.1}$$

   where A^2 represents the signal strength, and θ represents its noise variance. Accordingly, one can analyze different SNRs by varying the choice of K.

2. We set our detector to a constant nominal theta θ_0 based on prior knowledge.

3. We specify a value to the probability of false alarm constraint α.

4. We obtain our samples from a sinusoidal signal with unknown amplitude A.

B. Sampling of a Sinusoidal Signal With Unknown Amplitude

In this section, we derive the sampling technique used for the robustness analysis. Let $s(t)$ be a sinusoidal signal with unknown amplitude A, measured at time t, be defined by:

$$s(t) = A\cos(\omega t + \phi), \quad t = 0, \dots, T \tag{3.2}$$

where the period T of the sinusoidal signal is given by:

$$T = \frac{2\pi}{\omega} \tag{3.3}$$

Letting the phase shift $\phi = 0$, we obtain:

$$s(t) = A\cos(\omega t + 0) = A\cos(\omega t), \quad t = 0, \dots, T \tag{3.4}$$

Now lets sample this signal every Δt seconds, and do this i times, $i = 1, \dots, n$. The signal sample at time instance $t_i = i\Delta t$ can then be written as:

$$s_i = A\cos(\omega i \Delta t), \quad i = 1, \dots, n \tag{3.5}$$

Given that $w = 2\pi/T$, we can see that:

$$s_i = A\cos(2\pi \frac{\Delta t}{T} i), \quad i = 1, \dots, n \tag{3.6}$$

In order to simplify our expression, we can let $c_i = \cos\big(2\pi(\Delta t/T)i\big)$ and thus rewrite s_i as:

$$s_i = Ac_i, \quad i = 1, \dots, n \tag{3.7}$$

C. Threshold of Matched Filter Detector

In this section we are going to define the threshold T_r of our detector under nominal, noise only conditions, given the four model assumptions. Let $Y_i = N_i \sim N(0, \theta_0)$

under H_0 be a random sample, and let y_i be a realization of Y_i. Subsequently, the matched filter detector correlates our nominal signal s_{i_0} with y_i to create the test statistic $\sum_{i=1}^{n} s_{i_0} y_i$ used for making a decision based on the computed threshold T_r to either conclude H_0 or H_1. Thus, the optimal detection statistic can be defined as:

$$\sum_{i=1}^{n} s_{i_0} y_i \underset{H_0}{\overset{H_1}{\gtrless}} T_r \tag{3.8}$$

Knowing this, we can calculate the first moment of $\sum_{i=1}^{n} s_{i_0} y_i$ by:

$$E\Big\{\sum_{i=1}^{n} s_{i_0} Y_i\Big\} = \sum_{i=1}^{n} E\{s_{i_0} Y_i\} \tag{3.9}$$

$$= \sum_{i=1}^{n} s_{i_0} E\{Y_i\} \tag{3.10}$$

$$= \sum_{i=1}^{n} s_{i_0} \times 0 \tag{3.11}$$

$$= 0 \tag{3.12}$$

Similarly, the second moment can then also be calculated by:

$$E\Big\{(\sum_{i=1}^{n} s_{i_0} Y_i)^2\Big\} = E\Big\{\sum_{i=1}^{n}\sum_{j=1}^{n} s_{i_0} s_{j_0} Y_i Y_j\Big\} \tag{3.13}$$

$$= \Big\{\sum_{i=1}^{n}\sum_{j=1}^{n} s_{i_0} s_{j_0} \underbrace{E\{Y_i\}}_{=0} \underbrace{E\{Y_j\}}_{=0}\Big\}_{i\neq j} + \sum_{i=1}^{n} {s_{i_0}}^2 E\{{Y_i}^2\} \tag{3.14}$$

$$= \theta_0 \sum_{i=1}^{n} {s_{i_0}}^2 \tag{3.15}$$

Hence, under H_0:

$$\sum_{i=1}^{n} s_{i_0} Y_i \sim N(0, \theta_0 \sum_{i=1}^{n} {s_{i_0}}^2) = N(0, \theta_0 A_0^2 \sum_{i=1}^{n} {c_i}^2) \tag{3.16}$$

$$= N(0, K{\theta_0}^2 \sum_{i=1}^{n} {c_i}^2) \tag{3.17}$$

Using (3.8), we can write the probability of false alarm α of our detector, given H_0 under the nominal conditions as:

$$\alpha = P\Big(\sum_{i=1}^{n} s_{i_0} Y_i > T_r \mid Y_i \ = \text{ noise only}\Big) \tag{3.18}$$

$$= P\Big(N(0, K{\theta_0}^2 \sum_{i=1}^{n} {c_i}^2) > T_r\Big) \tag{3.19}$$

Next, using (2.2), we can represent the PDF of (3.19) as:

$$\alpha = \int_{T_r}^{\infty} N(0, K{\theta_0}^2 \sum_{i=1}^{n} {c_i}^2) \tag{3.20}$$

$$= \int_{T_r}^{\infty} \frac{1}{\sqrt{2\pi}\sqrt{K{\theta_0}^2 \sum_{i=1}^{n} {c_i}^2}} e^{\frac{-z^2}{2K{\theta_0}^2 \sum_{i=1}^{n} {c_i}^2}} dz \tag{3.21}$$

If we let $\omega = z/\sqrt{K{\theta_0}^2 \sum_{i=1}^{n} {c_i}^2}$, we can rewrite (3.21) to become:

$$\alpha = \int_{\frac{T_r}{\theta_0 \sqrt{K \sum_{i=1}^{n} {c_i}^2}}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{\frac{-\omega^2}{2}} d\omega \tag{3.22}$$

$$= \int_{\frac{T_r}{\theta_0 \sqrt{K \sum_{i=1}^{n} {c_i}^2}}}^{\infty} N(0, 1) \tag{3.23}$$

$$= \frac{1}{2} - erf\Big(\frac{T_r}{\theta_0 \sqrt{K \sum_{i=1}^{n} {c_i}^2}}\Big) \tag{3.24}$$

where we note $erf(x)$ to be defined by:

$$erf(x) = \int_0^x \frac{1}{\sqrt{2\pi}} e^{\frac{-\omega^2}{2}} d\omega \tag{3.25}$$

$$= \int_0^x N(0, 1) \tag{3.26}$$

$$= \frac{1}{2} - \int_x^{\infty} N(0, 1) \tag{3.27}$$

At last, using (3.24), we can solve for the threshold T_r constrained by α, resulting in:

$$T_r = \theta_0 \sqrt{K \sum_{i=1}^{n} {c_i}^2} \; erf^{-1}(\frac{1}{2} - \alpha) \tag{3.28}$$

D. Performance of Detector

At this point, we are done with the design of our detector under the nominal conditions, and are now going to divert our attention to calculating the detector's performance expressed as the detection probability β assuming non-nominal conditions. In order to determine this performance, we have to test our detector using H_1, where $Y_i = s_i + N_i \sim N(s_i, \sigma^2)$ is a random variable. As a result, we can define our detector as:

$$\sum_{i=1}^{n} s_{i_0} Y_i = \sum_{i=1}^{n} s_{i_0}(s_i + N_i) \tag{3.29}$$

Calculating the first moment of (3.29), we can see that:

$$E\Big\{ \sum_{i=1}^{n} s_{i_0} Y_i \Big\} = \sum_{i=1}^{n} E\Big\{ s_{i_0}(s_i + N_i) \Big\} \tag{3.30}$$

$$= \sum_{i=1}^{n} E\{s_{i_0} s_i\} + \sum_{i=1}^{n} E\{s_{i_0} N_i\} \tag{3.31}$$

$$= \sum_{i=1}^{n} E\{s_{i_0} s_i\} + 0 \tag{3.32}$$

$$= \sum_{i=1}^{n} s_{i_0} E\{s_i\} \tag{3.33}$$

$$= \sum_{i=1}^{n} A_0 c_i A c_i \tag{3.34}$$

$$= \sqrt{K\theta_0}\sqrt{K\theta} \sum_{i=1}^{n} c_i^2 \tag{3.35}$$

$$= K\sqrt{\theta_0 \theta} \sum_{i=1}^{n} c_i^2 \tag{3.36}$$

The second moment can be written as:

$$Var\Big\{\sum_{i=1}^{n} s_{i_0} Y_i\Big\} = Var\Big\{\sum_{i=1}^{n} s_{i_0}(s_i + N_i)\Big\} \tag{3.37}$$

$$= E\Big\{\big(\sum_{i=1}^{n} s_{i_0}(s_i + N_i) - \mu\big)^2\Big\} \tag{3.38}$$

$$= E\Big\{\big(\sum_{i=1}^{n} s_{i_0}(s_i + N_i) - K\sqrt{\theta_0\theta}\sum_{i=1}^{n} c_i^2\big)^2\Big\} \tag{3.39}$$

$$= E\Big\{\big(\sum_{i=1}^{n} s_{i_0}N_i + \sum_{i=1}^{n} s_{i_0}s_i - \sum_{i=1}^{n} s_{i_0}s_i\big)^2\Big\} \tag{3.40}$$

$$= E\Big\{\sum_{i=1}^{n} s_{i_0}^2 N_i^2\Big\} \tag{3.41}$$

$$= \sum_{i=1}^{n} E\{s_{i_0}^2\}E\{N_i^2\} \tag{3.42}$$

$$= \sum_{i=1}^{n} s_{i_0}^2\theta \tag{3.43}$$

$$= K\theta_0\theta\sum_{i=1}^{n} c_i^2 \tag{3.44}$$

Hence, under H_1:

$$\sum_{i=1}^{n} s_{i_0}Y_i \sim N(K\sqrt{\theta_0\theta}\sum_{i=1}^{n} c_i^2, K\theta_0\theta\sum_{i=1}^{n} c_i^2) \tag{3.45}$$

Next, using (3.8), we can define β as:

$$\beta = P(\sum_{i=1}^{n} s_{i_0}Y_i > T_r \mid Y_i \ = \ \text{signal} + \text{noise}) \tag{3.46}$$

$$= P\Big(N(K\sqrt{\theta_0\theta}\sum_{i=1}^{n} c_i^2, K\theta_0\theta\sum_{i=1}^{n} c_i^2) > T_r\Big) \tag{3.47}$$

Again, using (2.2), we can represent the PDF of (3.47) as:

$$\beta = \int_{T_r}^{\infty} N(K\sqrt{\theta_0\theta}\sum_{i=1}^{n} c_i^2, K\theta_0\theta\sum_{i=1}^{n} c_i^2) \tag{3.48}$$

$$= \int_{\frac{T_r - K\sqrt{\theta_0\theta}\sum_{i=1}^{n} c_i^2}{\sqrt{K\theta_0\theta\sum_{i=1}^{n} c_i^2}}}^{\infty} N(0,1) \tag{3.49}$$

$$= \int_{\frac{\theta_0\sqrt{K\sum_{i=1}^{n} {c_i}^2}\; erf^{-1}(\frac{1}{2}-\alpha) - K\sqrt{\theta_0\theta}\sum_{i=1}^{n} c_i^2}{\sqrt{K\theta_0\theta\sum_{i=1}^{n} c_i^2}}}^{\infty} N(0,1) \tag{3.50}$$

using T_r as defined in (3.28). Simplifying our lower integral limit, gives us:

$$\beta = \int_{\sqrt{\frac{\theta_0}{\theta}}erf^{-1}(\frac{1}{2}-\alpha) - \sqrt{K\sum_{i=1}^{n} c_i^2}}^{\infty} N(0,1) \tag{3.51}$$

E. Robustness of Detector

1. Performance Change

At this point, we have derived the probability of detection of our detector, and are interested in the detector's behavior when the measured θ changes. The most straightforward and common used method would be to calculate the slope of the tangent to the performance line at that point, which is done by taking the directional derivative of the detector's detection probability β with respect to θ. Applying this method gives us:

$$\frac{\partial\beta}{\partial\theta} = -\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\Big|_{y=\sqrt{\frac{\theta_0}{\theta}}erf^{-1}(\frac{1}{2}-\alpha) - \sqrt{K\sum_{i=1}^{n} c_i^2}} \cdot \tag{3.52}$$

$$\frac{\partial}{\partial\theta}\left(\sqrt{\frac{\theta_0}{\theta}}erf^{-1}(\frac{1}{2}-\alpha) - \sqrt{K\sum_{i=1}^{n} c_i^2}\right)$$

$$= -\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\sqrt{\frac{\theta_0}{\theta}}erf^{-1}(\frac{1}{2}-\alpha)-\sqrt{K\sum_{i=1}^{n}c_i^2})^2} \cdot (-\frac{1}{2})\frac{\theta_0^{\frac{1}{2}}}{\theta^{\frac{3}{2}}} erf^{-1}(\frac{1}{2}-\alpha) \tag{3.53}$$

$$= \frac{\sqrt{\theta_0} erf^{-1}(\frac{1}{2}-\alpha) e^{-\frac{1}{2}(\sqrt{\frac{\theta_0}{\theta}}erf^{-1}(\frac{1}{2}-\alpha)-\sqrt{K\sum_{i=1}^{n}c_i^2})^2}}{2\sqrt{2\pi}\theta^{\frac{3}{2}}} \tag{3.54}$$

2. Worst Case Slope to Riemannian Manifold

We previously determined the change of performance by taking the directional derivative of our performance function β. In order to consider the worst case used for a multi-dimensional manifold with regard to robustness analysis, we have to take the maximum directional derivative [8]. For example, in a two-dimensional manifold ($\mathbb{R}^3$ space), the curvature of a surface at a specific point does not have to be identical in all of the x, y, and z directions. Therefore, the maximum directional derivative at that point on the plane can be found by searching all directions, and choosing that particular direction that maximizes the tangent at this point. [8] derives the expression for the general worst case slope to a Riemannian manifold M using differential geometry techniques and formulates it as:

$$(D_{\vec{X}}h)^2 = \nabla h G^{-1} \nabla h^T \tag{3.55}$$

where $D_{\vec{X}}h$ represents the directional derivative of function h, ∇h is the gradient of h, G^{-1} is the inverse of the Riemannian metric g on the tangent space T_pM defined by:

$$g_{ij} = g\left(\frac{\partial}{\partial u_i}, \frac{\partial}{\partial u_j}\right) \tag{3.56}$$

$(u_i)_i$ being the local coordinates P on M.

As previously stated in (3.1), our analysis is based on a constant signal-to-noise ratio $K = A^2/\theta$; A^2 and θ varying dependently upon each other. Since we recognize K to be a continuous function, implies K is differentiable. Therefore, the tangent of

K is defined by the inner product:

$$
\begin{aligned}
g_{11} &= \langle V_1, V_1 \rangle && (3.57)\\
&= \|V_1\|^2 && (3.58)\\
&= (\Delta x)^2 + (\Delta y)^2 && (3.59)\\
&= (1)^2 + \Big(\frac{\Delta y}{\Delta x}\Big)^2 && (3.60)\\
&= 1 + \Big(\frac{dA}{d\theta}\Big)^2 && (3.61)\\
&= 1 + (\frac{1}{2}K^{\frac{1}{2}}\theta^{-\frac{1}{2}})^2 && (3.62)\\
&= 1 + \frac{K}{4\theta} && (3.63)
\end{aligned}
$$

Subsequently, we can rewrite (3.55) as:

$$
\begin{aligned}
Slope^2_{max} &= \frac{\partial\beta}{\partial\theta}\Big(1 + \frac{K}{4\theta}\Big)^{-1}\frac{\partial\beta}{\partial\theta} && (3.64)\\
&= \Big(1 + \frac{K}{4\theta}\Big)^{-1}\Big(\frac{\partial\beta}{\partial\theta}\Big)^2 && (3.65)\\
&= \Big(1 + \frac{K}{4\theta}\Big)^{-1} \cdot \frac{\theta_0\big(erf^{-1}(\frac{1}{2}-\alpha)\big)^2 e^{-\big(\sqrt{\frac{\theta_0}{\theta}}erf^{-1}(\frac{1}{2}-\alpha)-\sqrt{K\sum_{i=1}^{n}c_i^2}\big)^2}}{8\pi\theta^3} && (3.66)\\
&= \frac{\theta_0\big(erf^{-1}(\frac{1}{2}-\alpha)\big)^2 e^{-\big(\sqrt{\frac{\theta_0}{\theta}}erf^{-1}(\frac{1}{2}-\alpha)-\sqrt{K\sum_{i=1}^{n}c_i^2}\big)^2}}{2\pi(4\theta+K)\theta^2} && (3.67)
\end{aligned}
$$

resulting in:

$$
Slope(\theta) = \sqrt{\frac{\theta_0\big(erf^{-1}(\frac{1}{2}-\alpha)\big)^2 e^{-\big(\sqrt{\frac{\theta_0}{\theta}}erf^{-1}(\frac{1}{2}-\alpha)-\sqrt{K\sum_{i=1}^{n}c_i^2}\big)^2}}{2\pi(4\theta+K)\theta^2}} \qquad (3.68)
$$

where K, θ_0, α, and $c_i,\ i = 1, \ldots, n$ are fixed known constants.

To obtain an intuitive feeling of the derivations performed above, lets consider Fig. 2, where θ, A are plotted on the horizontal plane, and the third dimension β is plotted on the vertical plane. In this illustration, we observe β to be curving along

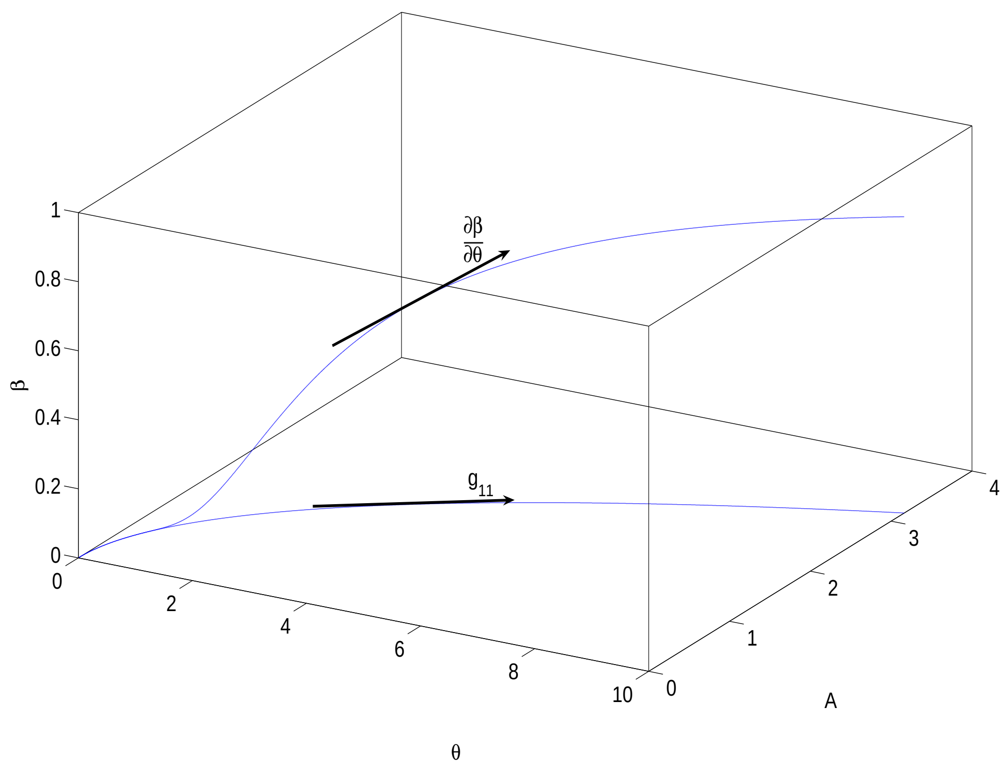

Fig. 2. The derivative components illustrated

with the horizontal plane defined by $K = A^2/\theta$. Consequently, if we wish to quantify the robustness in terms of slope, we need to split up the derivative of the performance curve β into two components:

1. The horizontal derivative, represented by g_{11}.

2. The vertical derivative, represented by $\partial\beta/\partial\theta$.

where we subsequently used (3.55) to provide us with the desired results derived earlier.

F. Robustness Analysis

The robustness quantification curves characterized for $n = 1$ - 20, containing signal strength $K = \{0.1, 1, 10\}$ with unit nominal variance are given in Figs. 3, 4, and 5. Additional curves for $n = 1$ - 20, containing signal strength $K = \{0.1, 1, 10\}$ with nominal variance $\theta_0 = \{0.1, 0.5, 2, 5, 10\}$ are located in Appendix A.

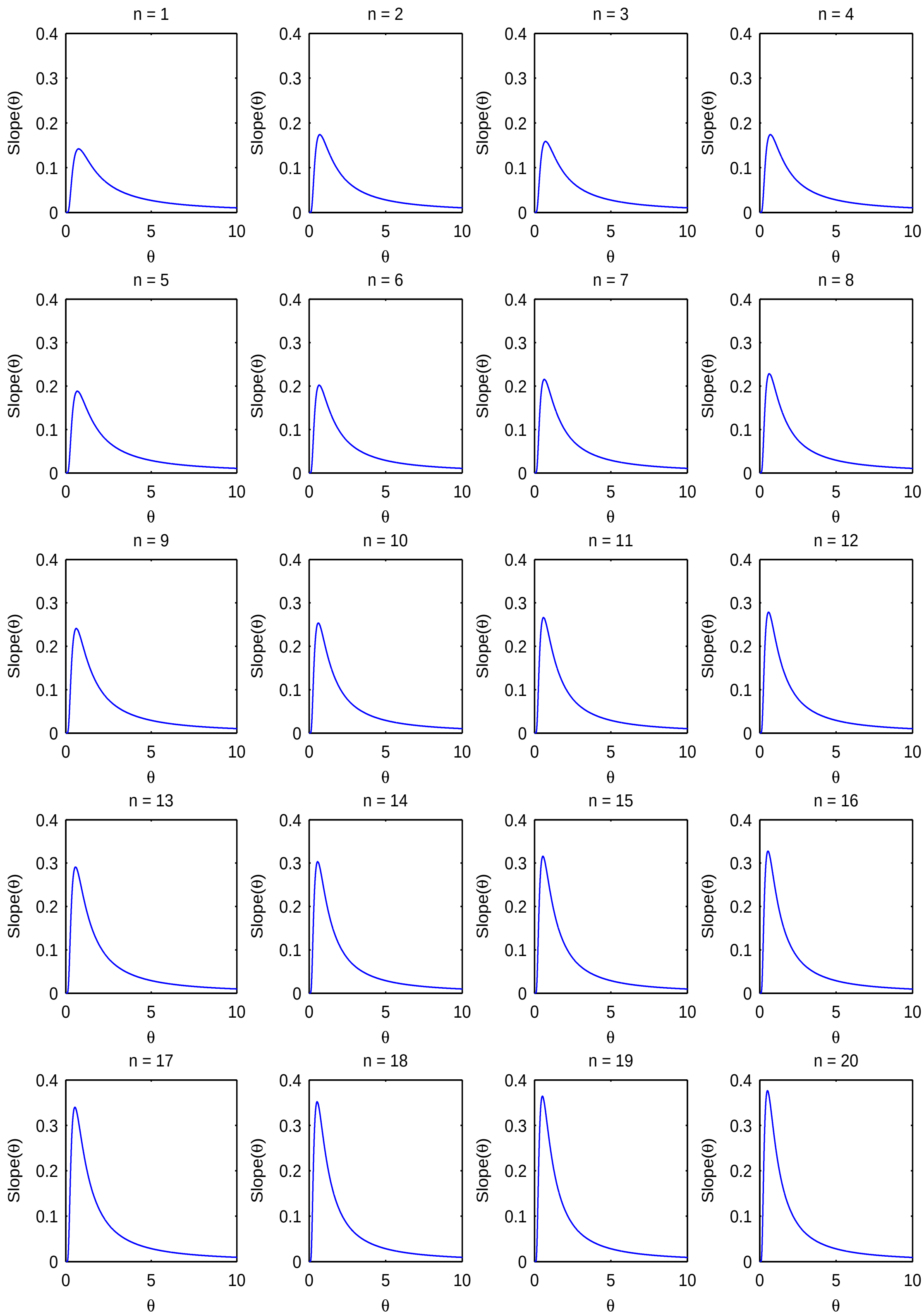

Fig. 3. $\big(\theta, Slope(\theta)\big)$ for $K = 0.1$, $\theta_0 = 1$

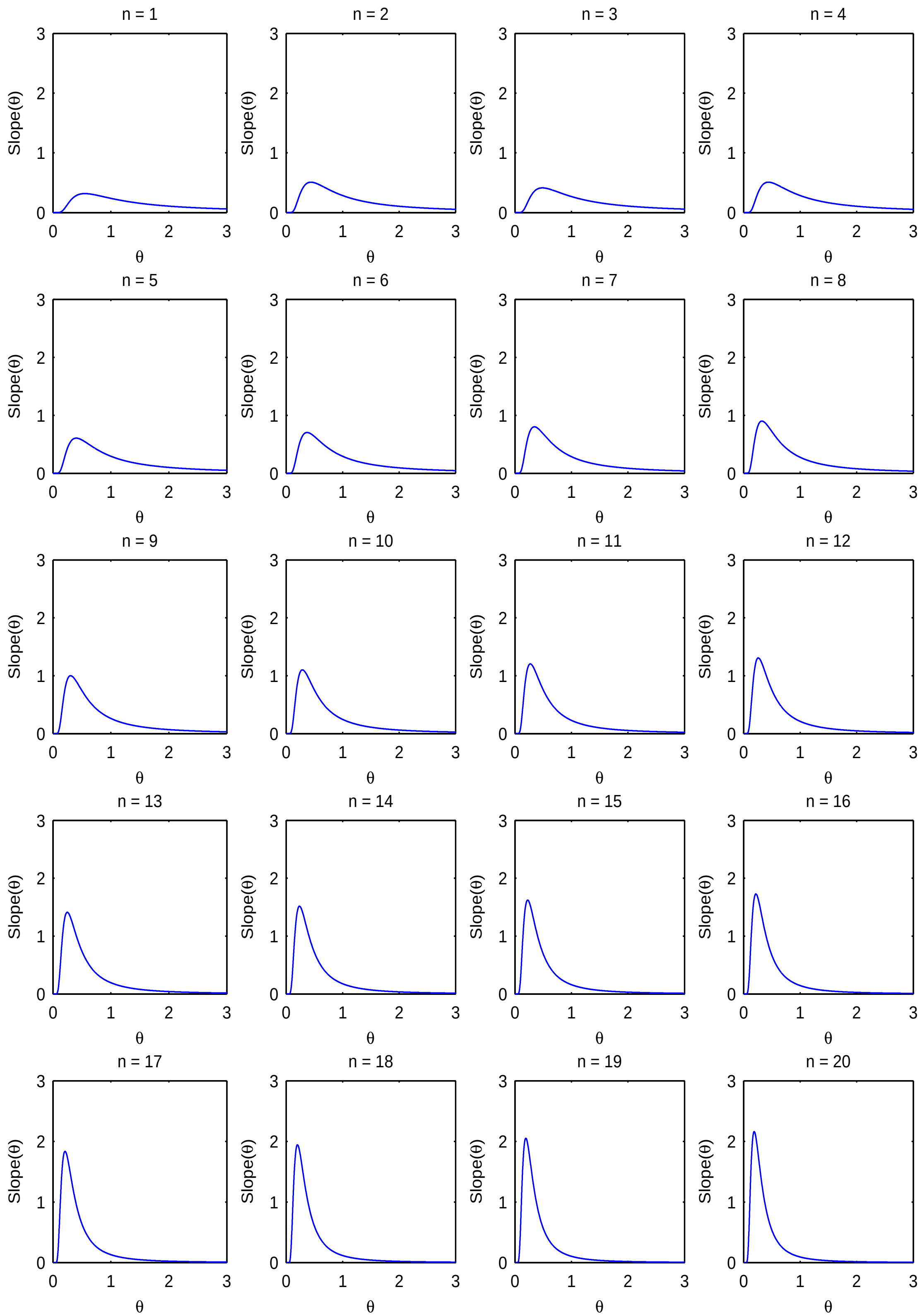

Fig. 4. $(\theta, Slope(\theta))$ for $K = 1$, $\theta_0 = 1$

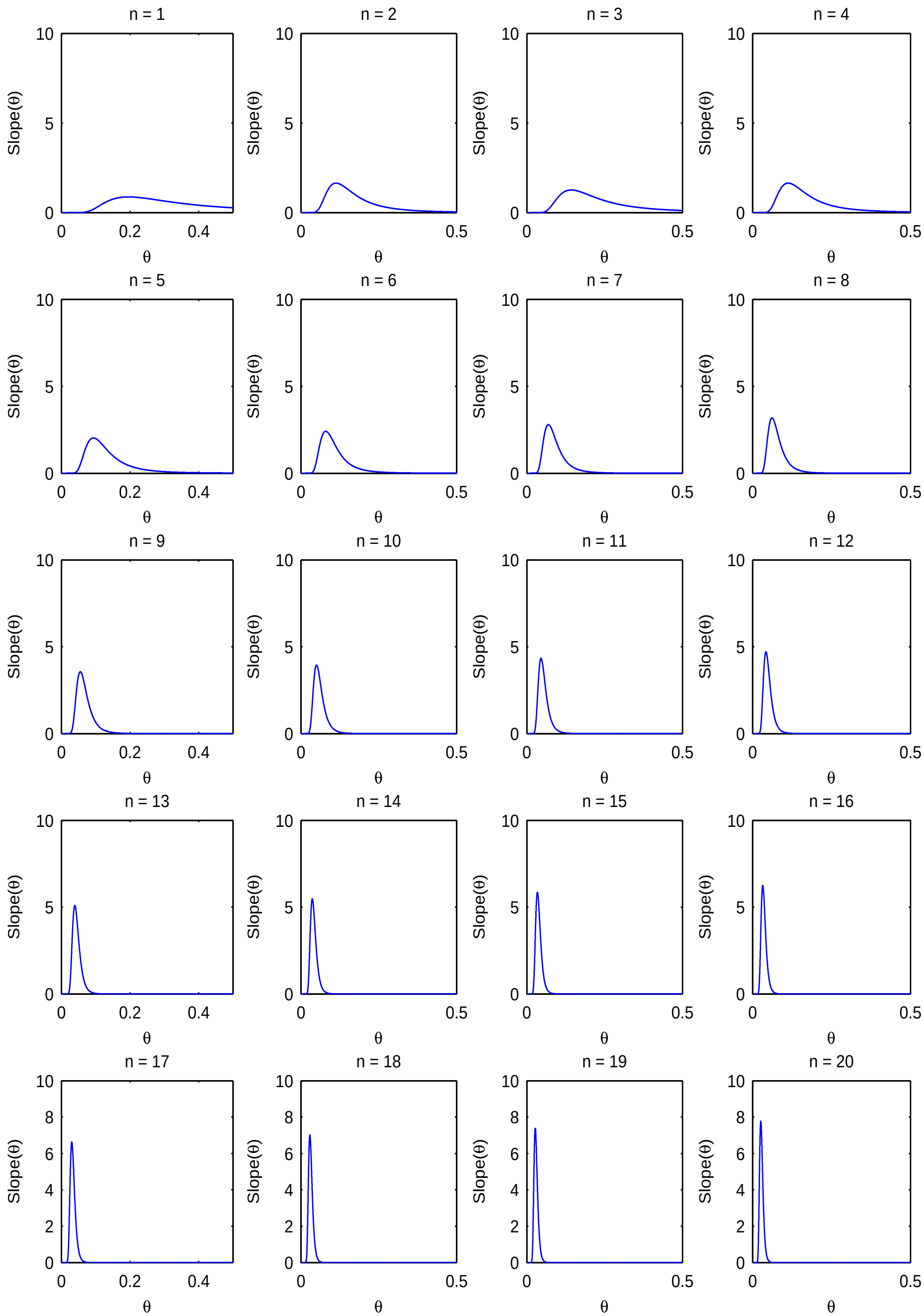

Fig. 5. $\big(\theta, Slope(\theta)\big)$ for $K = 10$, $\theta_0 = 1$

1. Common Signal Variance, $\theta \leq 10$

A first glance will reveal that when given a specific combination of K and θ_0, an increase in n results in an increasing the maximum slope as well. This phenomenon can be explained due to the fact that each of the n samples contains a measurement error, and since the noise is i.i.d., the total error for n samples is the product of each sample's individual error resulting in an increasing peak slope as n increases. Furthermore, we examined the consistency of this result for large n as well. Our findings were that for $n \geq 100$, the maximum slope monotonically increases for an increasing n. In addition, we observed that for $\theta_m \leq \theta \leq 10$, where $Slope(\theta_m) = Slope_{max}(\theta)$, $Slope(\theta)$ contains a faster decreasing downward slope for greater n. Clearly, these results go against our intuition of more samples taken being preferred to less in terms of the detector's performance and robustness.

Moreover, we note the peak slope at $Slope(\theta_m)$, given a fixed θ_0 and n, to increase for larger K. The cause for this rising peak is the high performance change $\Delta\beta$ for small θ. In other words, a small K results in a small $\Delta\beta$ and a large K results in a large $\Delta\beta$ with regard to small θ.

Another recognizable fact is that as θ_0 increases, given a particular K, the maximum slope decreases. This can be logically explained due to the fact that setting the detector to a higher θ_0, allows a greater range of plausible values for the detection of a signal, which in turn implies lesser performance, but greater robustness. Thus, in terms of robustness, high nominal variance is good. In terms of performance, high nominal variance is bad.

Another interesting point worth mentioning is the fact that for all K and θ_0, the peak of the slope is smaller at $n = 3$ compared to $n = 2$. This occurrence happens to be exceptional as the peak of $Slope(\theta)$ monotonically increases for all $n > 3$. We investigated this phenomenon and became to realize it was caused by the sampling method we employed, i.e., if we were to sample at different time instances within the designated period T, the peak of $Slope(\theta)$ would monotonically increase $\forall n$. Note that sampling at just any other random time instances within the designated period T will not always result in the peak of $Slope(\theta)$ to monotonically increase $\forall n$.

As of this moment, the results discussed above only hold for the case where we sample our signal within the limit of one period. In order for us to be able to conclude whether or not these results hold in general, we researched the effect of sampling beyond one period using the same sampling procedure as described in Chapter II. For example, we acquired n samples within the length of 5/4, 4/3, and 3/2 periods. Our findings were that there are no significant differences between sampling within or sampling beyond one period. Yet, in the case of sampling within the 3/2 periods, we found that when comparing $n = 3$ to $n = 2$, there exists an increase in the peak of the slope at $n = 3$ (special case previously stated) instead. However, this phenomenon does not have any influence on our overall results and will therefore be ignored in our research, making the above results valid in general.

2. Extraordinary Signal Variance, $\theta \geq 10$

The previous robustness analysis mainly focused on the common cases of the measured θ for a fixed θ_0, which is of main interest simply because this region contains the greatest performance change. However, in reality, some signals may contain both a large amplitude and variance, suggesting the detector has a "high" theoretical performance as can be seen in Fig. 2 (keep in mind the signal strength K remains unchanged). Thus, a vast underestimation of the real θ not anticipated on by the fixed nominal value may result in erroneous hypothesis decisions and non-optimal robustness situations. In this section, we investigate the effect of sample size at the design stage of the detector as it is the only parameter we are able to control in order to preemptively reduce hypothesis decision errors and improve robustness optimality.

a. Larger Sample Size Increases Decision Accuracy for Extraordinary Signal Variance

From the previous derivations in this chapter, we know that under

$$\begin{cases} H_0 : \sum_{i=1}^{n} s_{i0} Y_i \sim N(0, K{\theta_0}^2 \sum_{i=1}^{n} {c_i}^2) \\ H_1 : \sum_{i=1}^{n} s_{i0} Y_i \sim N(K\sqrt{\theta_0 \theta} \sum_{i=1}^{n} c_i^2, K\theta_0 \theta \sum_{i=1}^{n} c_i^2) \end{cases} \tag{3.69}$$

From (3.28) we obtained that the threshold $T_r = \theta_0 \sqrt{K \sum_{i=1}^{n} {c_i}^2}\ erf^{-1}(0.5 - \alpha)$. Suppose we have $K = 1$ and design our detector based on $\theta_0 = 5$ and $\alpha = 0.05$. For $n = 1$, we obtain $T_r \approx 8$. For $n = 20$, $T_r \approx 26$. Under ideal conditions, $\theta = \theta_0$. Hence, the threshold and the densities $f(x; H_0)$, $f(x; H_1)$ for $n = 1$ and $n = 20$ become as shown in Figs. 6 and 7. Clearly, when more samples are taken, we observe the decision surfaces to be less overlapping and the threshold to be less biased toward H_1. Despite only having a few samples, we can say this proposed model performs well when $\theta \approx \theta_0$. But what happens if $\theta \gg \theta_0$? Let's suppose the same scenario previously described, except let $\theta = 15$. The decision surfaces then become as shown in Figs. 8 and 9. Clearly, when only 1 sample is taken (Fig. 8), the large θ causes a significant overlap between $f(x; H_0)$ and $f(x; H_1)$. Thus, when strictly adhering to the calculated threshold T_r, we can recognize the fact that there exists a substantial chance of making the wrong decision exists, i.e., choosing the wrong hypothesis. However, when 20 samples are taken (Fig. 9), $f(x; H_0)$ and $f(x; H_1)$ are sufficiently separated to the point where the probability is adequately small for the value of x to reside in the same region of both densities. Furthermore, the threshold T_r makes a more intuitive distinction between the two hypothesis. Hence, when we

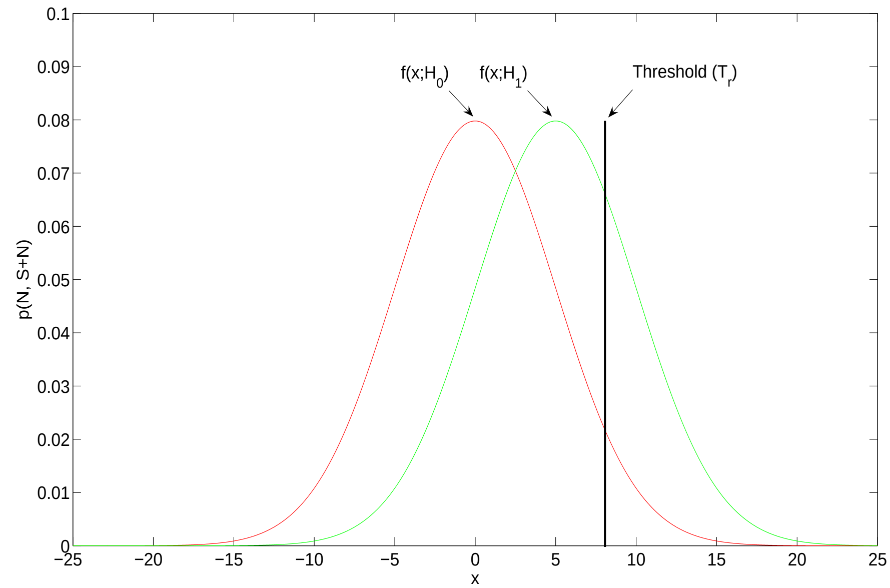

Fig. 6. Decision surface for $n = 1$ assuming $\theta = \theta_0$

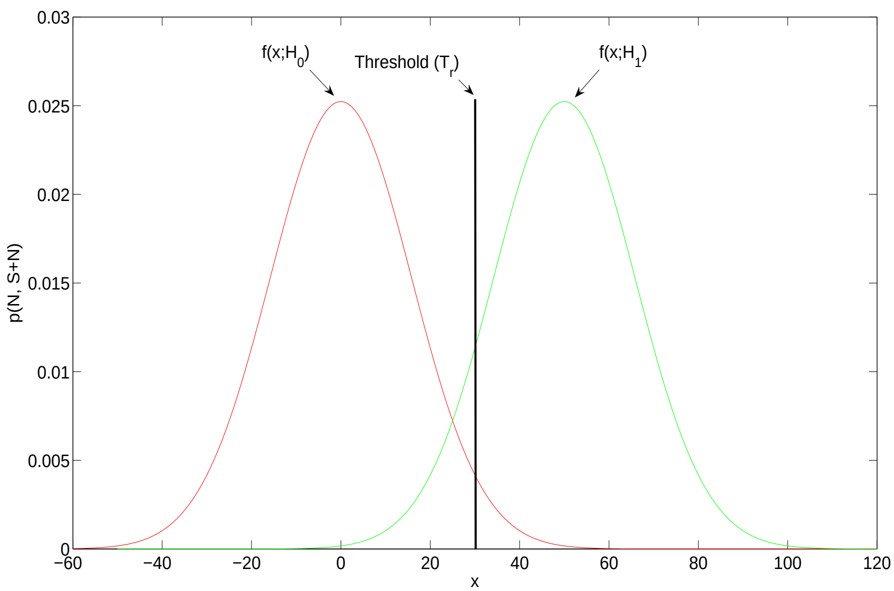

Fig. 7. Decision surface for $n = 20$ assuming $\theta = \theta_0$

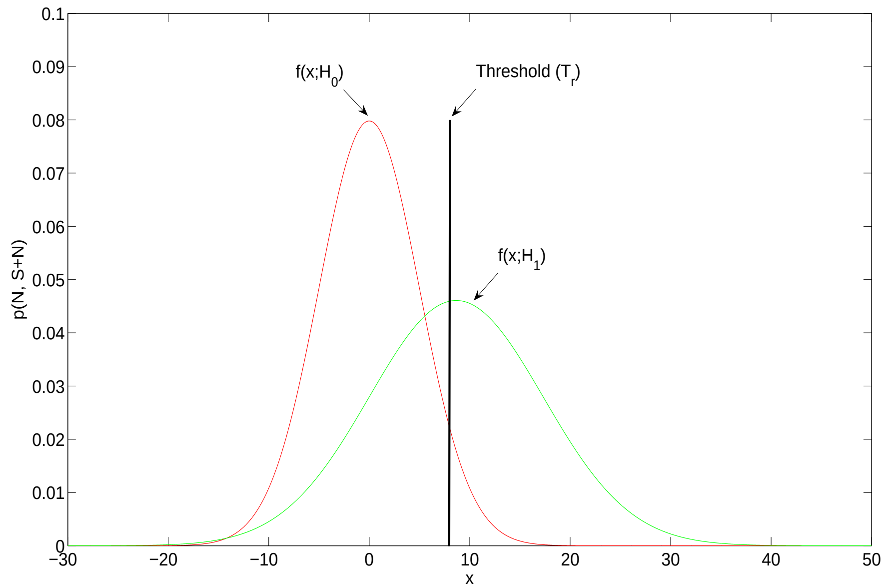

Fig. 8. Decision surface for $n = 1$ assuming $\theta_0 = 5$ and $\theta = 15$

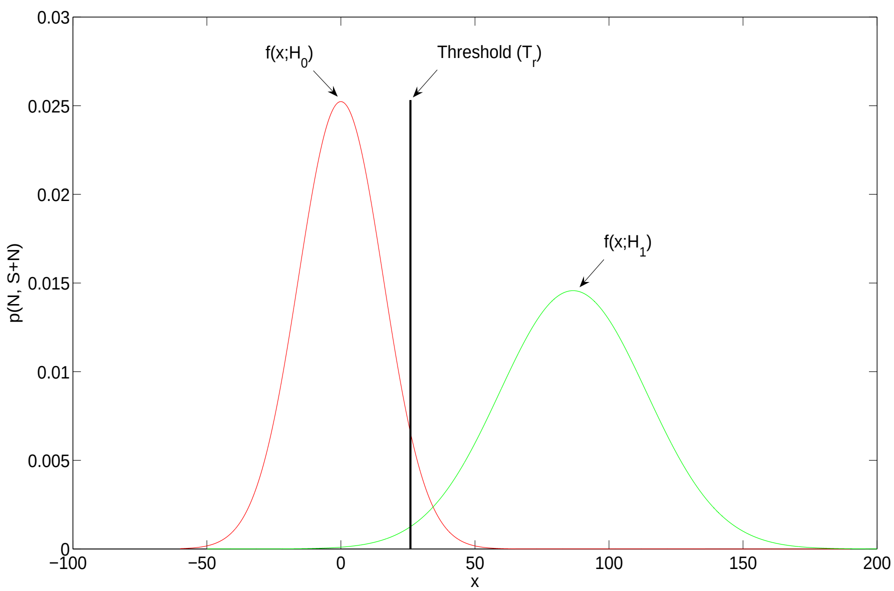

Fig. 9. Decision surface for $n = 20$ assuming $\theta_0 = 5$ and $\theta = 15$

are uncertain about the behavior of θ, taking more samples will enhance the ability of choosing the correct hypothesis.

b. Larger Sample Size Improves Robustness for Extraordinary Signal Variance

Let's once again refer to Figs. 3 - 5. Observing the shape of the graphs, we perceive that as θ increases, $Slope(\theta)$ seems to decrease exponentially. Therefore, we are interested in investigating the rate at which the tail of $Slope(\theta)$ decays. Suppose we model $Slope(\theta)$ to be of functional form:

$$Slope(\theta) = Ce^{-V\theta} \tag{3.70}$$

where C is a constant and V is the parameter of interest representing the rate at which the exponential decays. Taking the natural logarithm of both sides gives us:

$$\ln\bigl(Slope(\theta)\bigr) = \ln(C) - V\theta \tag{3.71}$$

Since we are interested in the rate of change (ROC) of $Slope(\theta)$ with respect to θ, we differentiate and obtain:

$$\frac{d}{d\theta}\ln\bigl(Slope(\theta)\bigr) = \frac{d}{d\theta}\ln(C) - \frac{d}{d\theta}V\theta \tag{3.72}$$

$$= -V \tag{3.73}$$

Hence, we can represent the ROC of the $\bigl(\theta, Slope(\theta)\bigr)$ curves over the interval $[\theta_a, \theta_b]$ for $\theta \geq 10$ by:

$$\text{ROC} = \frac{\ln\bigl(Slope(\theta_b)\bigr) - \ln\bigl(Slope(\theta_a)\bigr)}{\theta_b - \theta_a} \tag{3.74}$$

The ROC values of the interval $[10, 15]$ at $K = 0.1, 1, 10$ have been investigated and are shown in tables I - III.

We observe that for all combinations of K and θ_0 considered, the ROC decreases as n increases. Moreover, this result becomes more evident for larger K as the ROC values for a specific θ_0 decreases more rapid for an increasing n. Additionally, we perceive the actual tail values of $Slope(\theta)$ with $n > n_0, \forall n_0 \in \mathbb{N}$, to descend below the tail values of $Slope(\theta)$ with $n \leq n_0$, given a fixed θ_0 and K. Figs. 10 and 11 illustrate this concept. Recognizing the fact that both of these results hold in general, we can conclude that larger sample sizes will provide us with greater robustness for $\forall\theta \geq 10$.

Table I. ROC values of the interval $[10, 15]$ at $K = 0.1$

n	$\theta_0 = 0.1$	$\theta_0 = 1$	$\theta_0 = 2$	$\theta_0 = 5$	$\theta_0 = 10$
1	-0.1225635	-0.1185736	-0.1120542	-0.0899557	-0.050447
2	-0.1233543	-0.1210743	-0.1155907	-0.0955475	-0.058355
3	-0.1229926	-0.1199304	-0.1139731	-0.0929897	-0.0547378
4	-0.1233543	-0.1210743	-0.1155907	-0.0955475	-0.058355
5	-0.123673	-0.122082	-0.1170159	-0.0978009	-0.0615419
6	-0.1239611	-0.1229931	-0.1183044	-0.0998382	-0.064423
7	-0.1242261	-0.123831	-0.1194893	-0.1017116	-0.0670725
8	-0.1244727	-0.1246108	-0.1205922	-0.1034554	-0.0695385
9	-0.1247043	-0.1253433	-0.121628	-0.1050932	-0.0718547
10	-0.1249234	-0.126036	-0.1226077	-0.1066423	-0.0740454
11	-0.1251317	-0.1266949	-0.1235395	-0.1081156	-0.0761291
12	-0.1253308	-0.1273245	-0.1244299	-0.1095234	-0.07812
13	-0.1255218	-0.1279283	-0.1252839	-0.1108736	-0.0800295
14	-0.1257055	-0.1285094	-0.1261056	-0.1121729	-0.0818669
15	-0.1258828	-0.12907	-0.1268984	-0.1134265	-0.0836398
16	-0.1260543	-0.1296122	-0.1276653	-0.114639	-0.0853545
17	-0.1262204	-0.1301378	-0.1284085	-0.1158141	-0.0870163
18	-0.1263818	-0.1306481	-0.1291301	-0.1169551	-0.08863
19	-0.1265388	-0.1311444	-0.129832	-0.1180649	-0.0901995
20	-0.1266916	-0.1316278	-0.1305157	-0.1191458	-0.0917282

Table II. ROC values of the interval $[10, 15]$ at $K = 1$

n	$\theta_0 = 0.1$	$\theta_0 = 1$	$\theta_0 = 2$	$\theta_0 = 5$	$\theta_0 = 10$
1	-0.1259585	-0.1308946	-0.1297825	-0.1184127	-0.090995
2	-0.1284592	-0.1388026	-0.140966	-0.1360954	-0.1160022
3	-0.1273153	-0.1351853	-0.1358505	-0.128007	-0.1045634
4	-0.1284592	-0.1388026	-0.140966	-0.1360954	-0.1160022
5	-0.1294669	-0.1419894	-0.1454729	-0.1432214	-0.1260799
6	-0.130378	-0.1448706	-0.1495475	-0.1496638	-0.1351908
7	-0.1312159	-0.14752	-0.1532944	-0.1555883	-0.1435692
8	-0.1319957	-0.1499861	-0.156782	-0.1611026	-0.1513676
9	-0.1327282	-0.1523023	-0.1600575	-0.1662817	-0.158692
10	-0.1334209	-0.154493	-0.1631557	-0.1711803	-0.1656197
11	-0.1340798	-0.1565767	-0.1661024	-0.1758395	-0.1722087
12	-0.1347094	-0.1585676	-0.1689179	-0.1802912	-0.1785045
13	-0.1353133	-0.1604771	-0.1716184	-0.1845611	-0.184543
14	-0.1358943	-0.1623145	-0.1742169	-0.1886696	-0.1903533
15	-0.1364549	-0.1640873	-0.1767241	-0.1926339	-0.1959596
16	-0.1369972	-0.165802	-0.179149	-0.196468	-0.2013819
17	-0.1375227	-0.1674639	-0.1814993	-0.2001841	-0.2066373
18	-0.138033	-0.1690776	-0.1837814	-0.2037925	-0.2117402
19	-0.1385293	-0.1706471	-0.186001	-0.2073019	-0.2167033
20	-0.1390127	-0.1721757	-0.1881628	-0.2107201	-0.2215374

Table III. ROC values of the interval $[10, 15]$ at $K = 10$

n	$\theta_0 = 0.1$	$\theta_0 = 1$	$\theta_0 = 2$	$\theta_0 = 5$	$\theta_0 = 10$
1	-0.1329297	-0.1660928	-0.1820799	-0.2046371	-0.2154544
2	-0.1408377	-0.1910999	-0.2174453	-0.2605548	-0.294534
3	-0.1372205	-0.1796612	-0.2012685	-0.2349771	-0.2583616
4	-0.1408377	-0.1910999	-0.2174453	-0.2605548	-0.294534
5	-0.1440245	-0.2011777	-0.2316974	-0.2830893	-0.3264025
6	-0.1469057	-0.2102886	-0.2445822	-0.303462	-0.3552139
7	-0.1495552	-0.218667	-0.256431	-0.3221966	-0.3817087
8	-0.1520212	-0.2264654	-0.2674596	-0.3396344	-0.4063694
9	-0.1543374	-0.2337898	-0.277818	-0.3560123	-0.4295313
10	-0.1565281	-0.2407175	-0.2876151	-0.371503	-0.4514383
11	-0.1586118	-0.2473065	-0.2969334	-0.3862365	-0.4722748
12	-0.1606027	-0.2536023	-0.305837	-0.4003143	-0.4921838
13	-0.1625122	-0.2596408	-0.3143767	-0.4138167	-0.5112791
14	-0.1643496	-0.2654511	-0.3225938	-0.4268091	-0.529653
15	-0.1661225	-0.2710574	-0.3305222	-0.4393451	-0.5473817
16	-0.1678372	-0.2764797	-0.3381906	-0.4514698	-0.5645286
17	-0.169499	-0.2817351	-0.3456228	-0.4632212	-0.5811475
18	-0.1711127	-0.286838	-0.3528394	-0.4746317	-0.5972844
19	-0.1726822	-0.2918011	-0.3598582	-0.4857294	-0.612979
20	-0.1742109	-0.2966352	-0.3666947	-0.4965388	-0.6282657

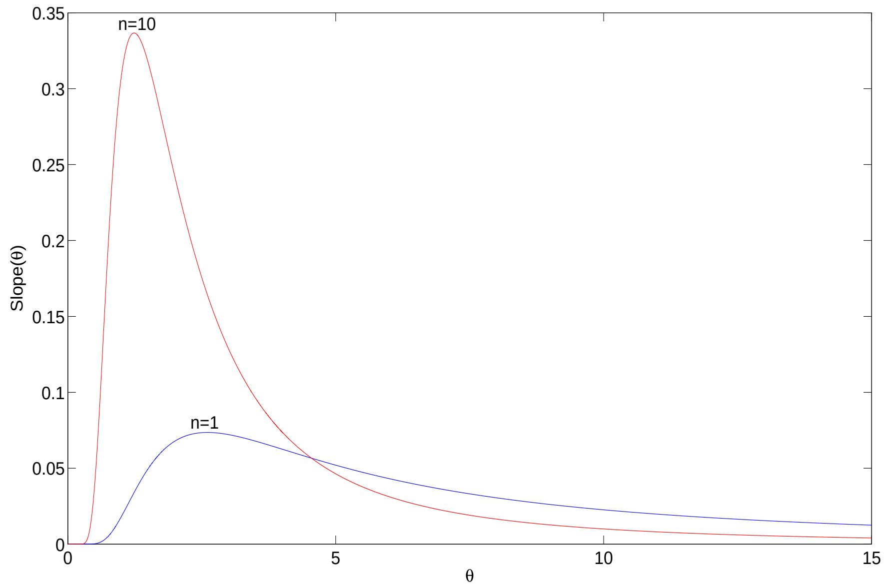

Fig. 10. $\big(\theta, Slope(\theta)\big)$ for $K = 1$ and $\theta_0 = 5$

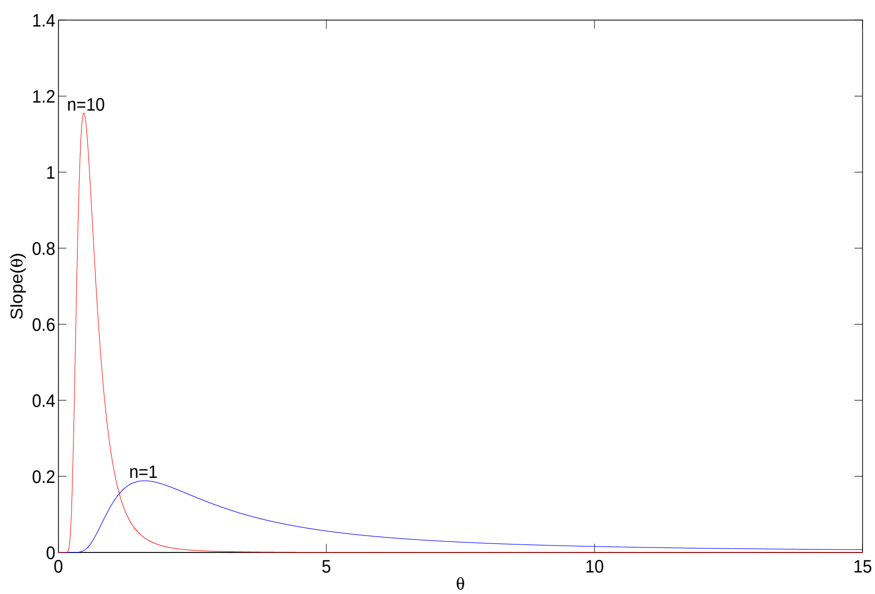

Fig. 11. $\big(\theta, Slope(\theta)\big)$ for $K = 4$ and $\theta_0 = 5$

CHAPTER IV

WEIGHTING METHOD APPROACH FOR ACHIEVING SLOPE LIKELIHOOD

In the previous chapter we utilized the slope method for quantifying the robustness of our detector for a variety of constant signal-to-noise ratios K and nominals θ_0. We are now going to expand our analysis in a way as such to give these slope values variable weights over a predetermined range of θ's. This chapter will present why there is a need for such a weighting method, the implementation details thereof, and the conclusion of our results.

A. Weighting Method Background

A weighting method is needed when some particular points on a curve are either over- or underrepresented. In our analysis, the need for applying a weighting method comes from the fact that all points on our $\big(\theta, Slope(\theta)\big)$ curvatures, plotted in Appendix A, are not equally probable. To recognize this fact, lets first get an intuitive feel of how (θ, A) and $Slope(\theta)$ are related to each other. Consider the three-dimensional Fig. 12, where (θ, A) is plotted on the horizontal plane, and $Slope(\theta)$ is plotted on the vertical plane. Given this illustration, we observe that $Slope(\theta)$ curves along with (θ, A), making it possible to quantify the robustness of our detector for a combination of θ and A.

Lets now refer our attention to Fig. 13, which is a two-dimensional representation of the horizontal plane of Fig. 12. Two intervals of equal length have been considered, namely $[\theta_1, \theta_2]$ and $[\theta_3, \theta_4]$. Let $\Delta\theta_{12}$, ΔA_{12} be the horizontal and vertical change of $[\theta_1, \theta_2]$, and let $\Delta\theta_{34}$, ΔA_{34} be the horizontal and vertical change of $[\theta_3, \theta_4]$. In addition, let dl_{12}, dl_{34} be the the length of the line segments of $[\theta_1, \theta_2]$, $[\theta_3, \theta_4]$ respectively. Since both $[\theta_1, \theta_2]$ and $[\theta_3, \theta_4]$ are of equal length, implies $\Delta\theta_{12} = \Delta\theta_{34}$. However,

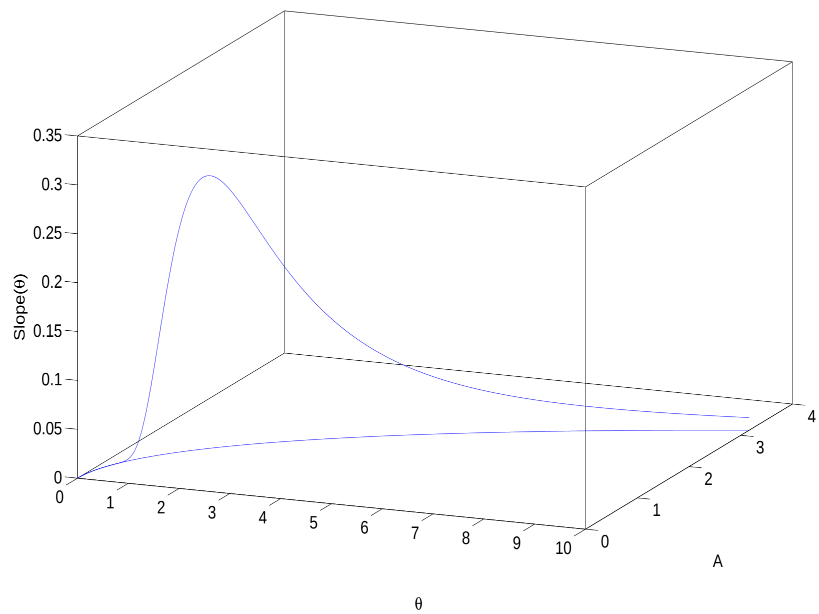

Fig. 12. Relationship between (θ, A) and $Slope(\theta)$

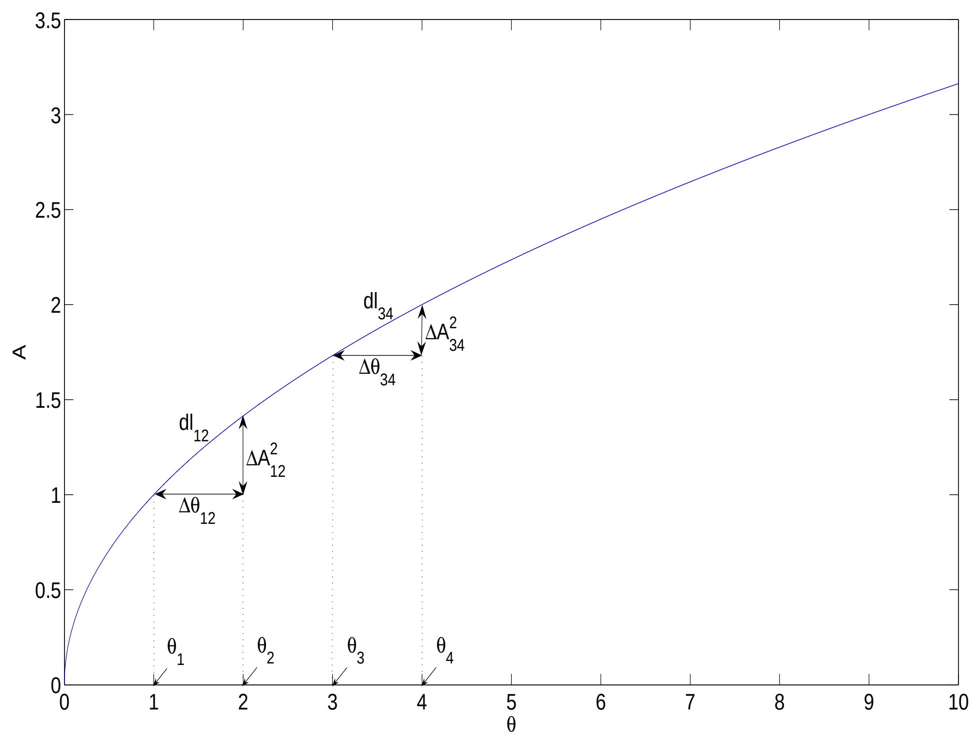

Fig. 13. Unequal length of line segments for smooth curvature

we can clearly perceive that $\Delta A_{12} \neq \Delta A_{34}$. Therefore, since the curve is sufficiently smooth logically implies $dl_{12} \neq dl_{34}$. As a result, all points on the curve (θ, A) are not equally probable, making each point on the $Slope(\theta)$ curve not equally probable as well. Therefore, applying a weighting factor would be the most uncomplicated and straightforward approach of dealing with these phenomena.

B. Implementation Details

For illustration purposes only, let Fig. 14 represent a typical graph for the quantification of robustness with θ plotted on the horizontal axis, and $Slope(\theta)$ plotted on the vertical axis. The first step of our weighting procedure is to evaluate the $Slope(\theta)$

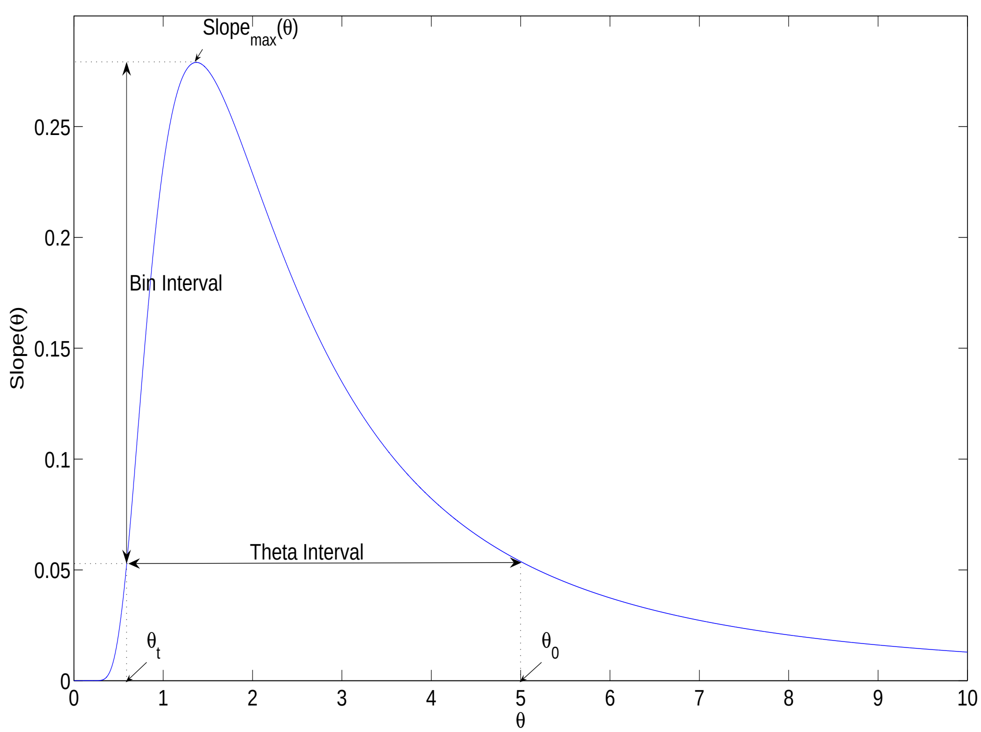

Fig. 14. Intervals used for weighting procedure

at $\theta = \theta_0$. Once this value has been determined, we need to find $\theta = \theta_t$, such that $Slope(\theta_0) = Slope(\theta_t)$. Since guaranteed convergence plays a more important role in our analysis than does rate of convergence, we used the Bisection Method [19] to find this $\theta = \theta_t$. Now having established the interval $[\theta_t, \theta_0]$, called the tolerable range, we can carry on with your procedure by equally parsing $[\theta_t, \theta_0]$ into 1000 theta values, implying the existence of 999 intervals.

One may wonder why we only choose the $[\theta_t, \theta_0]$ region for weighting the slopes? Our main motivation comes from the fact that this specific fragment contains the most commonly found θ values in conjunction with the uppermost slope values. In other words, this region contains the utmost changes of β leading to the worst case robustness scenarios of our detector.

Our next step is to divide the vertical axis from $Slope(\theta_t)$ to the maximum slope value, $Slope_{max}(\theta)$, into 100 equally sized bins. As will become more apparent later, 100 bins have been chosen in order to provide us with an average ratio of 10 theta values per bin, creating a sufficient environment for analysis.

Keeping Fig. 14 in the back of our mind, lets now focus our attention to Fig. 15, which is an exact replica of Fig. 13. As expected, we observe Fig. 15 to contain the same interval $[\theta_t, \theta_0]$ as Fig. 14, where the arc length of the curve [20], such as the arc length of $[\theta_{p5}, \theta_{p6}]$, are used for computing the equalizing factor. In our case, the arc length can be written as:

$$ds = \sqrt{(d\theta)^2 + (dA)^2}\ d\theta \tag{4.1}$$

where $\sqrt{(d\theta)^2 + (dA)^2}$ represents the equalizer. Factoring out $(d\theta)^2$ from the equalizer, we obtain:

$$equalizer = \sqrt{1 + \left(\frac{dA}{d\theta}\right)^2} \tag{4.2}$$

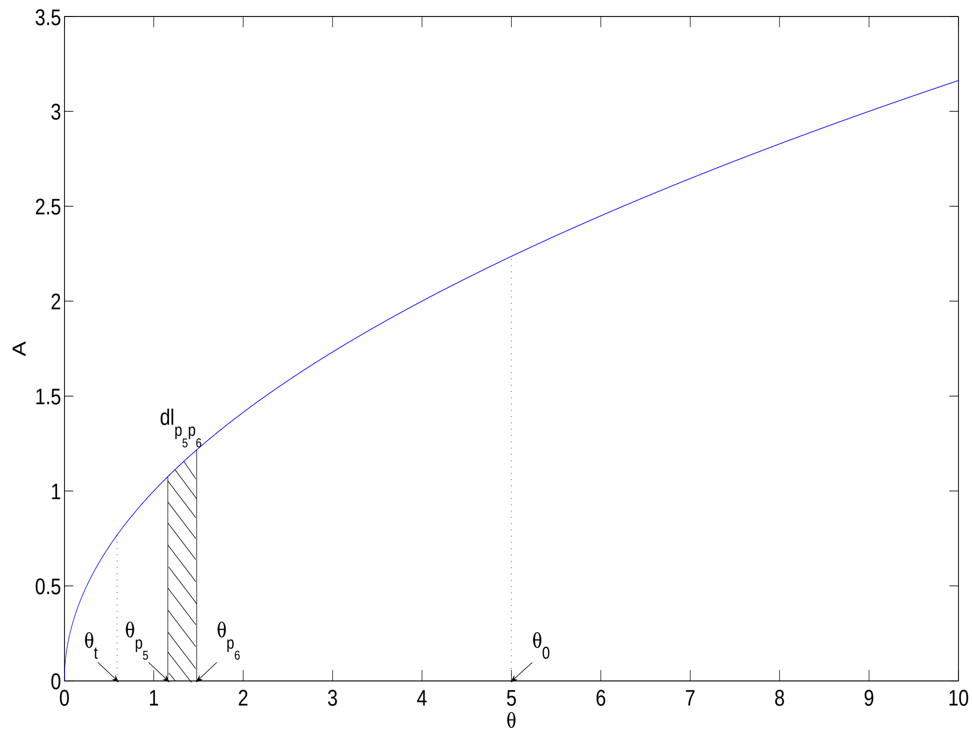

Fig. 15. Weighting procedure illustrated with respect to (θ, A) curvature for $[\theta_{p5}, \theta_{p6}]$

Referring back to (3.1), we can rewrite $K = A^2/\theta$ as:

$$A = \sqrt{K\theta} \tag{4.3}$$

Taking the derivative of A with respect to θ, gives us:

$$\frac{dA}{d\theta} = \frac{\sqrt{K}}{2\sqrt{\theta}} \tag{4.4}$$

Squaring (4.4) and substituting it into (4.1), we obtain:

$$equalizer = \sqrt{1 + \left(\frac{\sqrt{K}}{2\sqrt{\theta}}\right)^2} \tag{4.5}$$

$$= \sqrt{1 + \frac{K}{4\theta}} \tag{4.6}$$

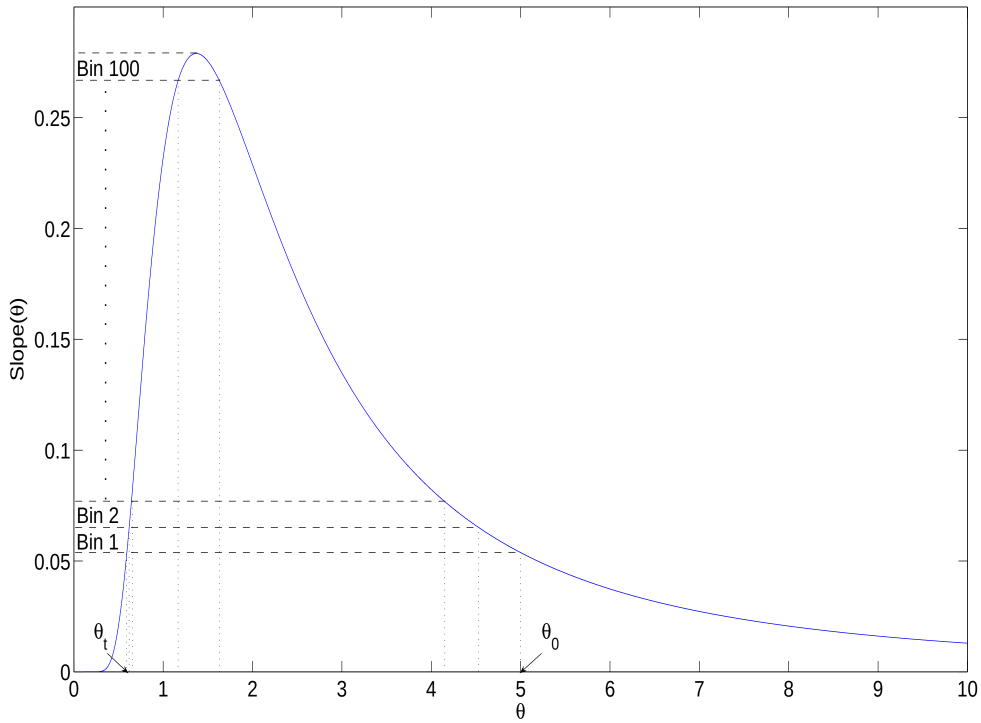

Fig. 16. Weighting procedure illustrated with respect to $\big(\theta, Slope(\theta)\big)$ curvature

Thus, the weight for the line segment on interval $[\theta_{p5}, \theta_{p6}]$, as illustrated in Fig. 15, can be written as:

$$weight_{p5p6} = \int_{\theta_{p5}}^{\theta_{p6}} \sqrt{1 + \frac{K}{4\theta}}\, d\theta \tag{4.7}$$

This weighting procedure is performed for each of the 1000 points on the interval $[\theta_t, \theta_0 + (\theta_0 - \theta_t)/999]$. Note that the weight for point θ_{p_5} corresponds to the interval $[\theta_{p_5}, \theta_{p_6}]$.

The final step of our weighting procedure involves the categorization of all the the weighted line segments into 1 of the 100 bins. This can be achieved by first searching which of the 1000 theta values on $[\theta_t, \theta_0]$ of the $Slope(\theta)$ curvature can be associated to bin 1 (Fig. 16). Once this has been accomplished, we proceed by placing their corresponding weighted line segments into the first bin, where for example, the value of θ_{p_5} corresponds to the weighted line segment $[\theta_{p_5}, \theta_{p_6}]$. Repeating this procedure 99 more times for the remaining 99 bins, we obtain *Likelihood* as a function of $Slope(\theta)$.

C. Slope Likelihood Analysis

The Slope Likelihood (SL) curves in histogram format for n = 1 - 20, containing signal strength $K = \{0.1, 1, 4\}$ with unit nominal variance are shown in Figs. 17, 18, and 19. Additional curves for n = 1 - 20, containing signal strength $K = \{0.1, 1, 4\}$ with nominal variance $\theta_0 = \{0.1, 0.5, 2, 5, 10\}$ are located in Appendix B.

Note that this procedure did not consider analyzing $K = 10$ as it resulted in numerous empty bins. Hence, creating a difficult situation for analysis. These empty bins are caused by the fact that a small increase in θ resulted in a large increase in $Slope(\theta)$, larger than the width of one bin. Therefore, a total of 100 bins is insufficient for analyzing the case where $K = 10$. We decided to perform our analysis for the case where $K = 4$ simply because it does not result in empty bins for all θ_0, $n \leq 20$. In addition, the validity of the analysis remains identical for both $K = 4$ and $K = 10$.

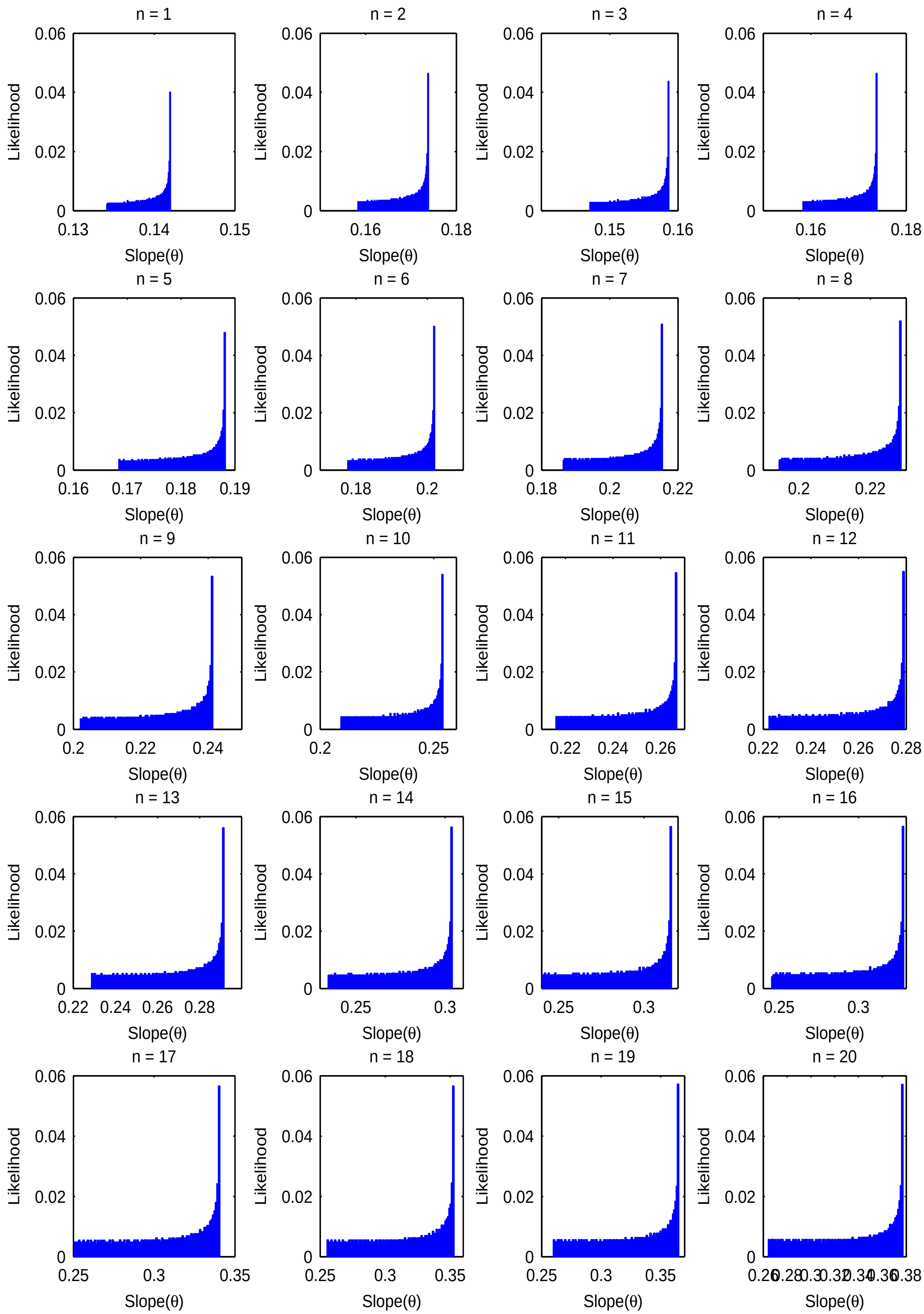

Fig. 17. Slope Likelihood for $K = 0.1$, $\theta_0 = 1$

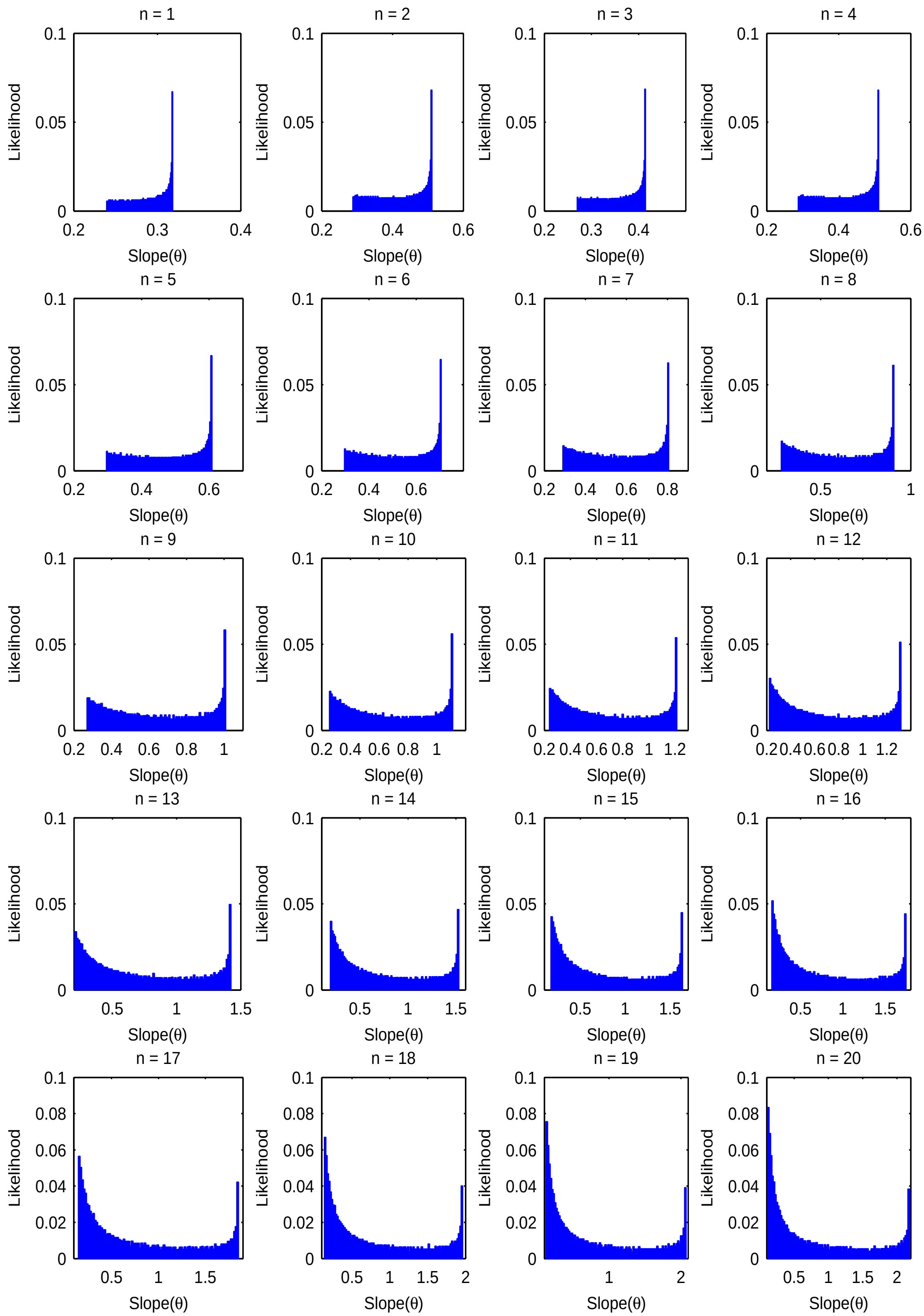

Fig. 18. Slope Likelihood for $K = 1$, $\theta_0 = 1$

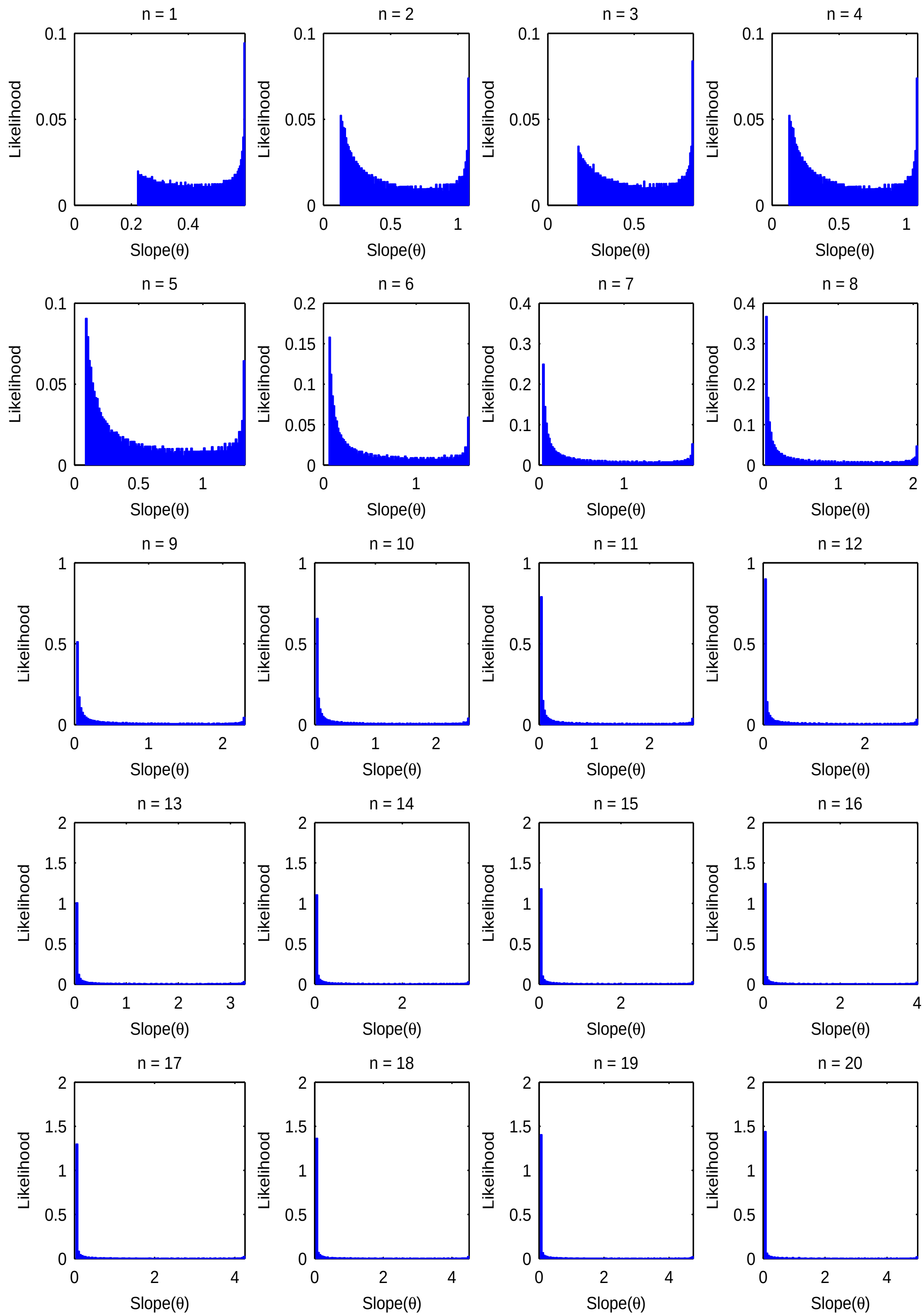

Fig. 19. Slope Likelihood for $K = 4$, $\theta_0 = 1$

At first, let's focus our attention to the horizontal axis $Slope(\theta)$ of the SL curves. A close look reveals the existence of an increasing slope range width as n increases, given a fixed K and θ_0. This result is directly related to the robustness quantification curves given in Figs. 3, 4, 5, and Appendix A, where we recognized the $Slope_{max}$ value to increase for an increasing n. However, the exception is once again made for the $n = 3$ case, where the width of the slope range decreases slightly when compared to the slope range of $n = 2$, but yet again increases monotonically for $n > 3$.

For $K = 0.1$, all n and θ_0 considered, the curves show a steady *Likelihood* increase for up to the first 75% of the $Slope(\theta)$ interval, followed by a strongly increasing *Likelihood* for the remaining 25%, resulting in a high peak at the end of the interval. Hence, when having a weak signal, the curves suggest there is a high likelihood for a high slope, making the detector unrobust.

For $K = 1$, all n and θ_0 considered, the curves again show a strong increase of the *Likelihood* near the end of the $Slope(\theta)$ interval. However, as n increases, the *Likelihood* of having a small $Slope(\theta)$ steadily increases as well. This phenomenon seems to become more apparent for smaller n as θ_0 increases. Hence, when having moderate signal strength, the curves suggest that the likelihood of obtaining greater robustness increases according to its sample size n, where this occurrence becomes more apparent for smaller n as θ_0 increases.

For $K = 4$ and all θ_0 considered, the curves show an increasing likelihood of having smaller slope values as n becomes greater. Generally, when $n \geq 6$, smaller slope values are more likely to occur. Hence, when dealing with a strong signal, the curves suggest that when the primary aim is to achieve greater robustness, more samples taken will be preferred to less.

CHAPTER V

QUANTITATIVE ROBUSTNESS ANALYSIS TECHNIQUES

In the preceding chapter we implemented a weighting method to insure all points on the $\big(\theta, Slope(\theta)\big)$ curvature are equally probable. We presented a basic analysis and made some general intuitive statements based upon the graphic results. However, the study lacks the quantitative component of the analysis. For that reason, this chapter will examine as to how likely we are to encounter the extreme slope values. Additionally, we will take a different approach that utilizes slope values to compute a confidence level of being robust based on some nominal delta performance constraint $\Delta\beta_0$.

A. Measuring Robustness in Terms of Likelihood

Based upon Figs. 17 - 19 and Figs. 39 - 53 located in Appendix B, we can generally see a pattern where the *Likelihood* fluctuates most at the first and final quarter of the tolerable range. The remaining 50% located around the center does not provide us with any additional relevant information, since the *Likelihood* on this interval remains approximately constant. We are therefore interested in analyzing the behavior of the SL at both of these outermost sections. Consequently, we can compute as to how likely the detector is to being robust by taking the ratio of these outermost areas A_1 and A_3 as illustrated in Fig. 20. Note that each rectangle represents a bin as discussed in Chapter IV. Therefore, given the fact that these bins of the SL curves are displayed in histogram format, we can easily compute the total area of an interval by adding up the areas of each individual rectangle. The likelihood ratio (LR) of the areas A_1/A_3 have been computed for the usual $K = \{0.1, 1, 4\}$, $\theta_0 = \{0.1, 0.5, 1, 2, 5, 10\}$, $n = 1 - 20$, and are shown in Tables IV - VI.

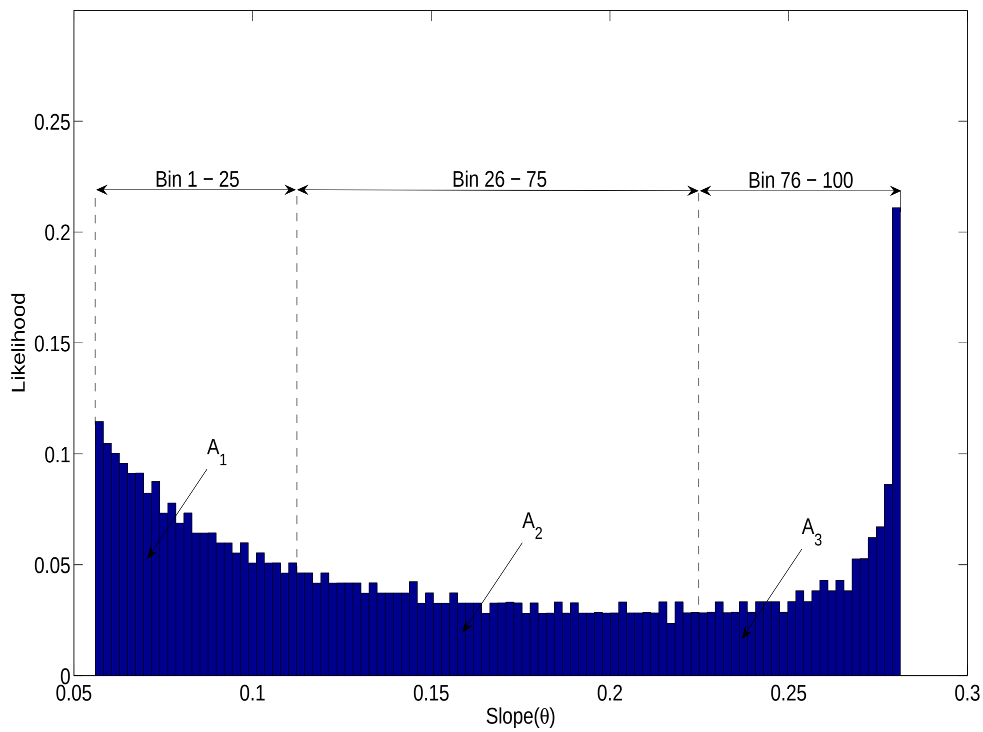

Fig. 20. Areas $\{A_1, A_2, A_3\}$ illustrated

Table IV. Likelihood ratio of areas A_1/A_3 at $K = 0.1$

n	$\theta_0 = 0.1$	$\theta_0 = 0.5$	$\theta_0 = 1$	$\theta_0 = 2$	$\theta_0 = 5$	$\theta_0 = 10$
1	0.282	0.294	0.295	0.297	0.294	0.295
2	0.295	0.307	0.310	0.309	0.311	0.312
3	0.288	0.297	0.304	0.305	0.303	0.307
4	0.295	0.307	0.310	0.309	0.311	0.312
5	0.299	0.316	0.322	0.320	0.323	0.325
6	0.310	0.323	0.327	0.332	0.332	0.330
7	0.310	0.334	0.336	0.339	0.339	0.335
8	0.321	0.339	0.343	0.344	0.348	0.349
9	0.327	0.346	0.352	0.353	0.354	0.357
10	0.330	0.355	0.361	0.365	0.366	0.361
11	0.336	0.364	0.369	0.372	0.370	0.371
12	0.343	0.368	0.376	0.381	0.380	0.381
13	0.350	0.376	0.388	0.387	0.388	0.390
14	0.357	0.383	0.393	0.397	0.398	0.400
15	0.361	0.395	0.402	0.407	0.407	0.405
16	0.367	0.402	0.409	0.417	0.413	0.418
17	0.373	0.412	0.421	0.422	0.424	0.424
18	0.380	0.424	0.426	0.429	0.435	0.435
19	0.389	0.426	0.436	0.443	0.441	0.447
20	0.392	0.435	0.444	0.451	0.450	0.453

Table V. Likelihood ratio of areas A_1/A_3 at $K = 1$

n	$\theta_0 = 0.1$	$\theta_0 = 0.5$	$\theta_0 = 1$	$\theta_0 = 2$	$\theta_0 = 5$	$\theta_0 = 10$
1	0.319	0.366	0.392	0.415	0.437	0.444
2	0.407	0.493	0.541	0.592	0.629	0.648
3	0.357	0.424	0.465	0.497	0.525	0.538
4	0.407	0.493	0.541	0.592	0.629	0.648
5	0.461	0.567	0.634	0.687	0.745	0.772
6	0.515	0.650	0.732	0.807	0.873	0.902
7	0.585	0.737	0.837	0.933	1.016	1.058
8	0.656	0.838	0.961	1.065	1.164	1.213
9	0.745	0.948	1.086	1.214	1.335	1.396
10	0.827	1.072	1.226	1.376	1.526	1.595
11	0.925	1.199	1.370	1.541	1.724	1.804
12	1.042	1.334	1.539	1.733	1.934	2.032
13	1.152	1.492	1.706	1.938	2.160	2.267
14	1.275	1.629	1.897	2.147	2.391	2.519
15	1.386	1.803	2.082	2.367	2.675	2.797
16	1.534	1.963	2.296	2.591	2.929	3.068
17	1.668	2.155	2.483	2.817	3.193	3.362
18	1.825	2.329	2.694	3.078	3.464	3.647
19	1.966	2.528	2.921	3.318	3.763	3.984
20	2.131	2.737	3.127	3.602	4.056	4.292

Table VI. Likelihood ratio of areas A_1/A_3 at $K = 4$

n	$\theta_0 = 0.1$	$\theta_0 = 0.5$	$\theta_0 = 1$	$\theta_0 = 2$	$\theta_0 = 5$	$\theta_0 = 10$
1	0.614	0.674	0.749	0.845	0.993	1.091
2	1.414	1.570	1.737	1.970	2.396	2.687
3	0.951	1.056	1.183	1.334	1.604	1.798
4	1.414	1.570	1.737	1.970	2.396	2.687
5	1.975	2.197	2.404	2.746	3.286	3.736
6	2.605	2.884	3.172	3.584	4.269	4.900
7	3.328	3.614	3.942	4.446	5.359	6.022
8	4.051	4.425	4.737	5.336	6.379	7.310
9	4.803	5.221	5.659	6.282	7.522	8.580
10	5.568	6.007	6.510	7.234	8.674	10.002
11	6.318	6.726	7.311	8.206	9.714	11.262
12	7.047	7.709	8.195	9.057	10.97	12.593
13	7.832	8.414	9.111	10.063	12.022	13.887
14	8.595	9.197	9.866	11.066	13.341	15.459
15	9.380	10.022	10.678	12.028	14.155	16.491
16	10.258	10.914	11.678	13.146	15.581	18.232
17	10.946	11.722	12.475	13.742	16.724	19.168
18	11.816	12.653	13.506	14.851	18.127	20.863
19	12.928	13.805	14.664	16.165	19.266	22.848
20	13.110	14.029	15.528	17.138	20.402	23.445

As a rule of thumb, we consider the detector to be rather robust if LR ≥ 2, i.e., if you are twice as likely to reside in the robust portion (area A_1) of the SL curve as opposed to the non-robust portion (area A_3). Conversely, we consider the detector to be non-robust if LR ≤ 0.5. However, when $0.5 <$ LR < 2, we are uncertain about the state of robustness and no conclusion can be drawn from the data. Note that we should not strictly adhere to these defined boundaries as they are not absolute.

Observing the results for $K = 0.1$, we perceive there exists an overall low LR for all θ_0, despite the existence of an increasing LR for greater n. However, for $K = 1$, the LR increases considerably faster as n increases, where this occurrence becomes more evident for larger θ_0. When $K = 10$, the same conclusion can be drawn in a more extreme fashion, i.e., the LR increases significantly faster for an increasing n, where this again becomes more apparent for larger θ_0.

Conclusively, the data from Tables IV - VI suggests the detector becomes more robust for larger K, θ_0, and n.

B. Measuring Robustness in Terms of Confidence Level Given Constraint $\Delta\beta(\theta_0)$

1. Implementation Details

In general applications, a detector or algorithm is considered robust if the performance change $\Delta\beta$ does not deviate by more than 5% from the expected performance, i.e., we do not want the performance to change by more than 5% from β_0. Hence, we need:

$$\Delta\beta_0 \leq (0.05)\beta_0 \tag{5.1}$$

$$= (0.05)\beta(\theta_0) \tag{5.2}$$

In Chapter III, we formulated a direct relationship between the performance change of the detector and the $(\theta, Slope(\theta))$ curvatures. Fig. 21 illustrates this fact where $(\theta, Slope(\theta))$ is plotted on the horizontal plane and the performance β is plotted on the vertical axis. As a consequence of this direct relationship, we can upper bound

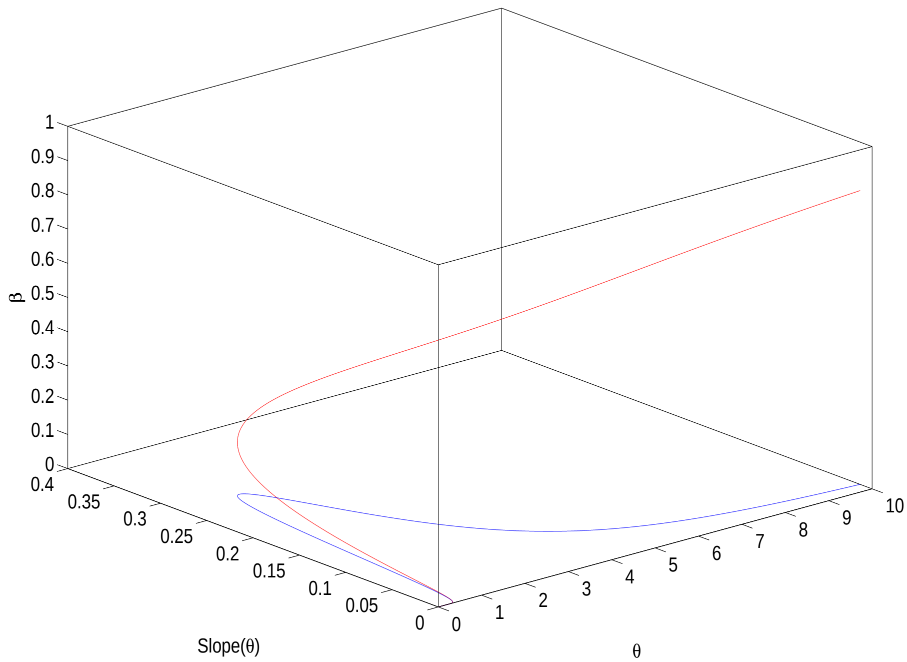

Fig. 21. Relationship between $(\theta, Slope(\theta))$ and β illustrated

the performance change $\Delta\beta$ in terms of θ and $Slope(\theta)$ by:

$$\Delta\beta \leq \big(\text{maximum variation in } \theta\big)\big(\text{maximum variation in } Slope(\theta)\big) \tag{5.3}$$

$$= (\theta_0 - \theta_c)Slope(\theta_c) \tag{5.4}$$

$\forall\theta_c \in \mathbb{R}^+$. Subsequently, given the fact that we do not want to allow β to change by more than 5% from the nominal, we require:

$$(\theta_0 - \theta_c)Slope(\theta_c) \leq (0.05)\beta(\theta_0) \tag{5.5}$$

Thus in order calculate the maximum distance θ is allowed to travel from the nominal, we need to find $\theta = \theta_c$, such that:

$$(\theta_0 - \theta_c)Slope(\theta_c) = (0.05)\beta(\theta_0) \tag{5.6}$$

In the previous chapters we identified the least robust range of θ to be for $\theta \in [\theta_t, \theta_0]$. Therefore, as illustrated in Fig. 22, we are only interested in finding $\theta = \theta_c$ for $\theta_c \in [\theta_t, \theta_0]$. Note that when $\theta_c \leq \theta_m$, where $Slope(\theta_m) = Slope_{max}(\theta)$, we let

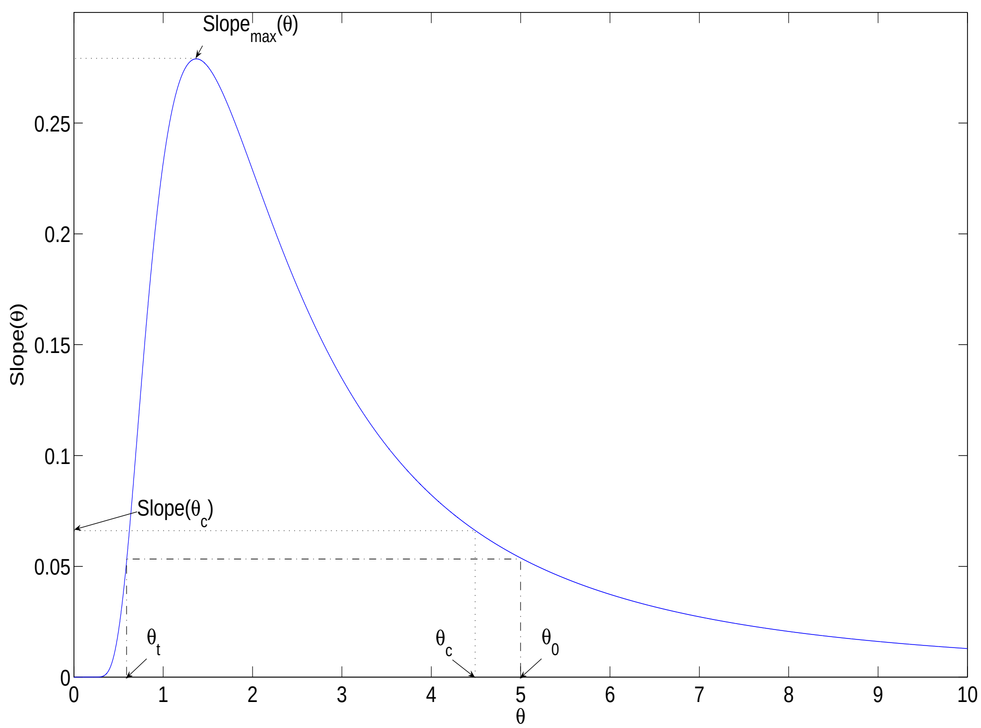

Fig. 22. Example of $\theta_c \in [\theta_t, \theta_0]$ satisfying the $(5\%)\beta(\theta_0)$ constraint

$Slope(\theta_c) = Slope(\theta_m), \forall \theta_c \leq \theta_m$.

Once we obtain $\theta = \theta_c$ for $\theta \in [\theta_t, \theta_0]$, we can express a confidence level percentage, %C, as a measure of robustness to be:

$$\%\mathrm{C} = \frac{\theta_0 - \theta_c}{\theta_0 - \theta_t} \cdot 100\% \tag{5.7}$$

where we assume θ to travel uniformly over the region $[\theta_t, \theta_0]$. Notice that a smaller θ_c will provide us with a greater level of confidence since it is allowed to travel further away from the nominal, without violating the $(5\%)\beta(\theta_0)$ constraint.

As previously stated, the method for computing the confidence level given by (5.7) assumes θ travels uniformly over the region $[\theta_t, \theta_0]$. However, since we have knowledge of a nominal value, it would be more realistic for θ_c to reside in the neighborhood of θ_0. We should therefore view θ as traveling from θ_0 to be probabilistic rather than direct. Hence, we need to apply a weighting factor that puts more emphasis on values closer to the nominal as opposed to values far away. Fig. 23 provides an illustration of such a weighting method as it weights according to the ratio of the areas under the triangle. More specifically, we first draw a straight line from the points $(\theta_t, 0)$ to $\big(\theta_0, Slope(\theta_0)\big)$, creating a triangular shape. Next, we obtain the value θ_c by applying the same approach previously described. Subsequently, we use this value to partition the triangle into two parts and take the ratio of the area to the right side of θ_c over the area of the complete triangle. Hence, we can write the weighted confidence percentage, %WC, as:

$$\%\mathrm{WC} = \frac{A_{tri}(\theta_0 - \theta_c)}{A_{tri}(\theta_0 - \theta_t)} \cdot 100\% \tag{5.8}$$

where $A_{tri}(A, B)$ denotes the area under the triangle from point A to point B located on the horizontal axis θ.

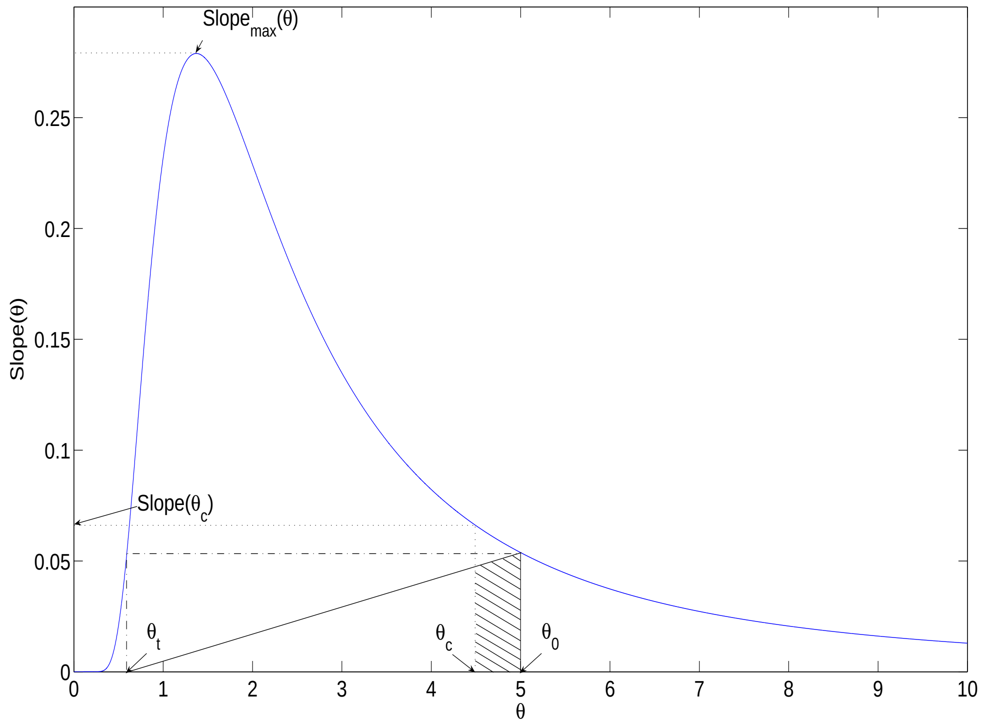

Fig. 23. Measuring weighted confidence illustrated

2. Results

The confidence (%C) and weighted confidence (%WC) computational results for $n = 1$ - 20, containing signal strength $K = \{0.1, 1, 10\}$ with unit nominal variance are given in Tables VII, VIII, and IX. Additional tables for $n = 1$ - 20, containing signal strength $K = \{0.1, 1, 10\}$ with nominal variance $\theta_0 = \{0.1, 0.5, 2, 5, 10\}$ are located in Appendix C.

As expected, the results confirm the weighted confidence values for all K, θ_0, and n to be greater as the non-weighted confidence values. Moreover, we would expect the confidence level to increase when more samples are taken. The data confirms our intuition for moderate ($K = 1$) and strong ($K = 10$) signal strengths. However, for

Table VII. Confidence procedure results for $K = 0.1$ and $\theta_0 = 1$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.0920	4.5981e-3	0.5902	0.9664	0.1358	8.2082	15.7427
2	0.1155	5.7750e-3	0.5156	0.9646	0.1609	7.3073	14.0806
3	0.1042	5.2125e-3	0.5482	0.9652	0.1492	7.7077	14.8213
4	0.1155	5.7750e-3	0.5156	0.9646	0.1609	7.3073	14.0806
5	0.1261	6.3052e-3	0.4888	0.9637	0.1716	7.1071	13.7091
6	0.1363	6.8130e-3	0.4660	0.9626	0.1814	7.0070	13.5230
7	0.1461	7.3040e-3	0.4462	0.9617	0.1905	6.9069	13.3368
8	0.1556	7.7819e-3	0.4287	0.9611	0.1990	6.8068	13.1503
9	0.1650	8.2492e-3	0.4130	0.9606	0.2071	6.7067	12.9636
10	0.1741	8.7075e-3	0.3988	0.9597	0.2148	6.7067	12.9636
11	0.1832	9.1581e-3	0.3859	0.9588	0.2222	6.7067	12.9636
12	0.1920	9.6020e-3	0.3740	0.9586	0.2290	6.6066	12.7767
13	0.2008	1.0040e-2	0.3631	0.9579	0.2357	6.6066	12.7767
14	0.2094	1.0472e-2	0.3529	0.9572	0.2421	6.6066	12.7767
15	0.2180	1.0900e-2	0.3434	0.9566	0.2482	6.6066	12.7767
16	0.2265	1.1323e-2	0.3346	0.9560	0.2540	6.6066	12.7767
17	0.2348	1.1741e-2	0.3262	0.9548	0.2598	6.7067	12.9636
18	0.2431	1.2156e-2	0.3184	0.9543	0.2652	6.7067	12.9636
19	0.2513	1.2566e-2	0.3110	0.9538	0.2703	6.7067	12.9636
20	0.2595	1.2973e-2	0.3040	0.9533	0.2753	6.7067	12.9636

Table VIII. Confidence procedure results for $K = 1$ and $\theta_0 = 1$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.2595	1.2973e-2	0.3390	0.9484	0.2492	7.8078	15.0060
2	0.4087	2.0437e-2	0.2362	0.9342	0.3097	8.6086	16.4761
3	0.3371	1.6857e-2	0.2774	0.9414	0.2858	8.1081	15.5588
4	0.4087	2.0437e-2	0.2362	0.9342	0.3097	8.6086	16.4761
5	0.4745	2.3727e-2	0.2061	0.9269	0.3237	9.2092	17.5703
6	0.5347	2.6734e-2	0.1831	0.9190	0.3302	9.9099	18.8378
7	0.5893	2.9467e-2	0.1648	0.9114	0.3305	10.6106	20.0954
8	0.6387	3.1935e-2	0.1498	0.9030	0.3264	11.4114	21.5206
9	0.6831	3.4154e-2	0.1374	0.8938	0.3192	12.3123	23.1087
10	0.7228	3.6138e-2	0.1268	0.8838	0.3098	13.3133	24.8542
11	0.7581	3.7905e-2	0.1178	0.8737	0.2985	14.3143	26.5796
12	0.7894	3.9472e-2	0.1100	0.8628	0.2864	15.4154	28.4545
13	0.8171	4.0857e-2	0.1031	0.8510	0.2740	16.6166	30.4721
14	0.8415	4.2076e-2	0.0970	0.8391	0.2611	17.8178	32.4609
15	0.8629	4.3147e-2	0.0916	0.8272	0.2480	19.0190	34.4208
16	0.8817	4.4084e-2	0.0867	0.8135	0.2361	20.4204	36.6709
17	0.8980	4.4902e-2	0.0824	0.8007	0.2238	21.7217	38.7251
18	0.9123	4.5615e-2	0.0784	0.7860	0.2131	23.2232	41.0533
19	0.9247	4.6234e-2	0.0748	0.7722	0.2021	24.6246	43.1855
20	0.9354	4.6770e-2	0.0715	0.7574	0.1922	26.1261	45.4265

Table IX. Confidence procedure results for $K = 10$ and $\theta_0 = 1$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.9354	4.6770e-2	0.0765	0.6700	0.1416	35.7357	58.7010
2	0.9977	4.9883e-2	0.0391	0.4008	0.0831	62.3624	85.8341
3	0.9871	4.9353e-2	0.0519	0.5198	0.1024	50.6507	75.6464
4	0.9977	4.9883e-2	0.0391	0.4008	0.0831	62.3624	85.8341
5	0.9996	4.9980e-2	0.0313	0.3154	0.0726	70.6707	91.3979
6	0.9999	4.9997e-2	0.0260	0.2551	0.0659	76.4765	94.4664
7	1.0000	5.0000e-2	0.0222	0.2102	0.0632	80.7808	96.3062
8	1.0000	5.0000e-2	0.0194	0.1774	0.0606	83.8839	97.4027
9	1.0000	5.0000e-2	0.0172	0.1530	0.0576	86.1862	98.0918
10	1.0000	5.0000e-2	0.0154	0.1337	0.0553	87.9880	98.5571
11	1.0000	5.0000e-2	0.0140	0.1176	0.0558	89.4895	98.8953
12	1.0000	5.0000e-2	0.0128	0.1057	0.0522	90.5906	99.1146
13	1.0000	5.0000e-2	0.0118	0.0949	0.0529	91.5916	99.2930
14	1.0000	5.0000e-2	0.0109	0.0862	0.0522	92.3924	99.4212
15	1.0000	5.0000e-2	0.0102	0.0785	0.0531	93.0931	99.5229
16	1.0000	5.0000e-2	0.0095	0.0730	0.0478	93.5936	99.5896
17	1.0000	5.0000e-2	0.0089	0.0675	0.0472	94.0941	99.6512
18	1.0000	5.0000e-2	0.0084	0.0620	0.0521	94.5946	99.7078
19	1.0000	5.0000e-2	0.0080	0.0586	0.0449	94.8949	99.7394
20	1.0000	5.0000e-2	0.0075	0.0542	0.0508	95.2953	99.7787

a weak signal ($K = 0.1$), we observe the confidence levels to actually decrease with greater n. This interesting development is caused by the fact that for all n, θ_c remains close to the nominal, while the tolerable range $[\theta_t, \theta_0]$ increases with a rising n. On the contrary, θ_c is more flexible to travel for moderate and strong signal strengths, resulting in larger confidence levels for greater n. Note that for larger SNRs, the confidence levels elevate more rapid as n increases.

For all K and θ_0 considered, we perceive all the values for $n = 2$ and $n = 4$ to be identical, including the confidence levels. Additionally, an exceptional decrease in the confidence level can be found for $n = 3$ when compared to $n = 2$. However, this is conversely true for $K = 0.1$ where there actually exists an increase in the confidence level for $n = 3$ over $n = 4$. The cause of this phenomenon carries the same argument as stated in chapter III, namely, the time instance t_i at which the samples are obtained. Essentially, this phenomenon would not have happened if different time instances within the designated period T would have been chosen for obtaining the samples. However, once again note that this does not appear to be the case when samples are taken at all other time instances within the designated period T.

CHAPTER VI

SUMMARY AND CONCLUSION

A. Summary

This thesis performed a quantitative analysis on the robustness of the matched filter detector corrupted by zero mean, independent and identically distributed white Gaussian noise. The notion of slope was used to analyze the robustness under different signal-to-noise ratios, nominals and sample sizes. The analysis of these slopes were divided into two parts, namely, slope for common signal variances and slope for extraordinary signal variances. Next, we proceeded with our research by applying a weighting method to the slope range of interest, the so called tolerable range, in order to measure the likelihood of these slopes to occur. Subsequently, we used the area residing in the first and last quarter section of this tolerable range to analyze the likelihood of achieving low slope values. Lastly, we developed a method that uses confidence as a measure of robustness. Both weighted and non-weighted procedures were applied over the tolerable range, where the weighted procedure assigned a heavier bias for values located near the nominal.

B. Conclusion

1. Weak Signal Strength

Lets first focus on the most common type of signal variances. In this case, larger data samples are detrimental as the value of the peak slope rises with larger n. Additionally, greater sample sizes reduces the confidence of residing within the maximum performance change bounds $\Delta\beta(\theta_0)$. Choosing a larger nominal on the other hand, does decrease the peak slope given any fixed sample size n. Unfortunately, this does not provide us with a helpful solution as the sacrifice of performance has to be taken into consideration. Although larger data samples does somewhat increase the likelihood ratio of residing in the lower slope range, it is not nearly enough to consider the detector robust, regardless of which nominal is chosen.

With regard to the extraordinary signal variances, we observe larger data samples to actually increase the robustness due to the exponential decay rate. This result is also dependent on which nominal is chosen; a smaller nominal results in a greater exponential decay rate. Since the decision surfaces greatly overlap in this case, taking more samples is trivial as it further separates the decision surfaces.

Overall, we conclude the matched detector to be non-robust in situations that deal with weak signal strengths. The most accurate results can be achieved by choosing the nominal to the best of your knowledge and by taking a moderate amount of samples in order to find a balance between detector performance and robustness.

2. Moderate Signal Strength

Lets again first put our attention to the most common type of signal variances. In this case, larger data samples causes the peak of the slopes to increase. On the other hand, larger sample sizes does induce greater confidence and likelihood ratios. The most noticeable effect of a greater nominal value in conjunction with larger sample sizes is the incremental rate at which the likelihood ratio increases.

Regarding extraordinary signal variances, the exponential decay rate becomes greater for larger sample sizes and nominals. However, nothing particularly interesting occurs concerning the exponential decay rate for low nominal values.

In general, we conclude the matched detector to be robust for moderate signal strengths if the nominal is not largely overestimated and when a reasonable amount of samples are taken. However, note that sample size should be adjusted according to the practitioner's certainty of having chosen the correct nominal value.

3. Strong Signal Strength

Even for strong signals, a high peak narrow band slope exists. However, the confidence levels and the likelihood ratios show there exists a slim chance of residing in these respective areas. In addition, few samples are needed for achieving high probability of detection. Generally, 6 samples will be sufficient for achieving good performance and robustness. For extremely strong signal, even fewer samples will suffice. When the case arises that the practitioner is not certain about the chosen nominal value, it would be of best interest to take some extra samples as to decrease the likelihood of residing in the non-robust high slope region. However, if this solution is too costly, it would be best to slightly underestimate the nominal in order to gain some confidence and performance despite the contradictory results shown by the likelihood ratio test.

C. Recommendations for Future Research

The most intuitive continuation of the above performed analysis is to somewhat relax the Gaussian assumption while retaining independence and stationarity. If such analysis proves to be successful, then further extension to dependence and non-stationarity could be pursued.

REFERENCES

[1] S. A. Kassam and H. V. Poor, “Robust techniques for signal processing: A survey,” *Proc. IEEE*, vol. 73, pp. 433–480, 1985.

[2] P.J. Huber, *Robust Statistics*, Wiley, New York, 1981.

[3] P. J. Huber, “Robust estimation of a local parameter,” *Ann. Math. Stat.*, vol. 35, pp. 73–101, 1964.

[4] H.-C. Lee and D. R. Halverson, “Robust estimator design with an emphasis balance between performance and robustness,” *J. Franklin Inst.*, vol. 344, no. 7, pp. 954–968, October 2007.

[5] M. W. Thompson and D. R. Halverson, “A differential geometric approach toward robust signal detection,” *J. Franklin Inst.*, vol. 328, no. 4, pp. 379–401, 1991.

[6] F. W. Kellison and D. R. Halverson, “Applications of curvature toward the measurement of robustness for data processors,” *Computational Statistics and Data Analysis*, vol. 37, no. 3, pp. 343–362, 2001.

[7] C. Tsai and D. R. Halverson, “Average robustness in signal detection and estimation,” *J. Franklin Inst.*, vol. 333 B, no. 1, pp. 127–139, 1996.

[8] D. R. Halverson and V. Varma, “Applications of unbiased perturbations towards quantifying robustness with pragmatic geometric methods,” *Computational Statistics and Data Analysis*, vol. 49, no. 4, pp. 1228–1243, 2005.

[9] W. Liu and D. R. Halverson, “Robustness of the sign detector in dependent noise,” *J. Franklin Inst.*, vol. 336, pp. 1155–1173, September 1999.

[10] J. J. Fuchs, "A robust matched detector," *IEEE Trans. on Signal Processing*, vol. 55, no. 11, pp. 5133–5142, November 2007.

[11] M. N. Desai and R. S. Mangoubi, "Robust gaussian and non-gaussian matched subspace detection," *IEEE Trans. on Signal Processing*, vol. 51, no. 12, pp. 3115–3127, December 2003.

[12] A. Dogandžić and B. Zhang, "Bayesian complex amplitude estimation and adaptive matched filter detection in low-rank interference," *IEEE Trans. on Signal Processing*, vol. 55, no. 3, pp. 1176–1182, March 2007.

[13] H. V. Poor, *An Introduction to Signal Detection and Estimation*, Springer-Verlag, New York, NY, 2nd edition, 1994.

[14] R. D. Hippenstiel, *Detection Theory: Applications and Digital Signal Processing*, CRC Press, Boca Raton, FL, 1st edition, 2002.

[15] G. Casella and R. L. Berger, *Statistical Inference*, Duxbury Press, Pacific Grove, CA, 2nd edition, 2001.

[16] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*, Prentice Hall, Upper Saddle River, NJ, 2000.

[17] H. L. V. Trees, *Detection, Estimation, and Modulation Theory, Part I*, John Wiley and Sons, New York, NY, 1968.

[18] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, vol. I, Prentice Hall, Upper Saddle River, NJ, 1993.

[19] S. D. Conte and C. de Boor, *Elementary Numerical Analysis: An Algorithmic Approach*, McGraw-Hill, New York, NY, 3rd edition, 1980.

[20] G. L. Raux, "Robustness measures for signal detection in non-stationary noise using differential geometric tools," Ph.D. dissertation, Texas A&M University, 2006.

APPENDIX A

ROBUSTNESS QUANTIFICATION CURVES

Robustness quantification curves characterized by $Slope(\theta)$ for $n = 1$ - 20, $K = \{0.1, 1, 10\}$, and $\theta_0 = \{0.1, 0.5, 2, 5, 10\}$ are given in Figs. 24 - 38.

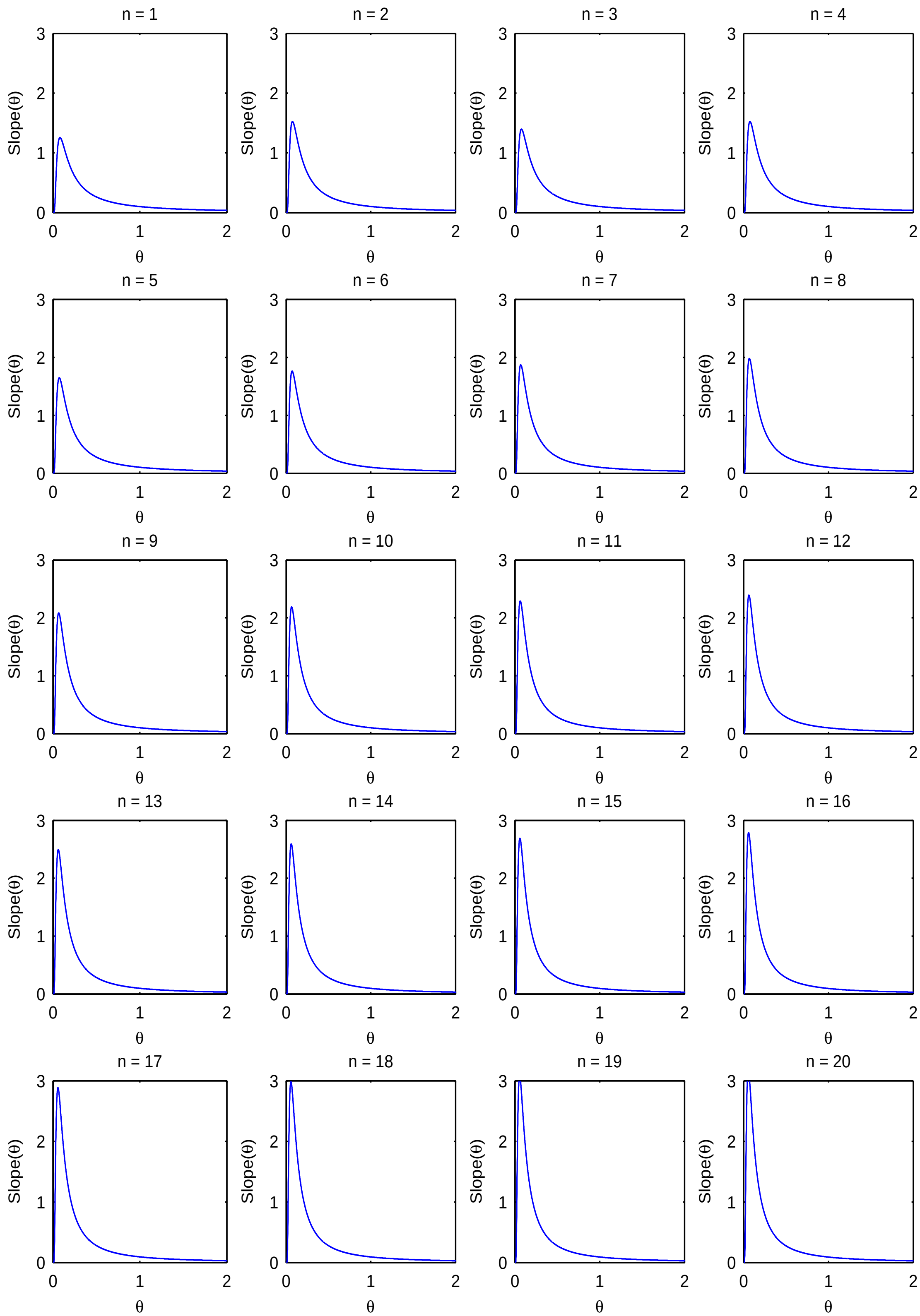

Fig. 24. $\big(\theta, Slope(\theta)\big)$ for $K = 0.1$, $\theta_0 = 0.1$

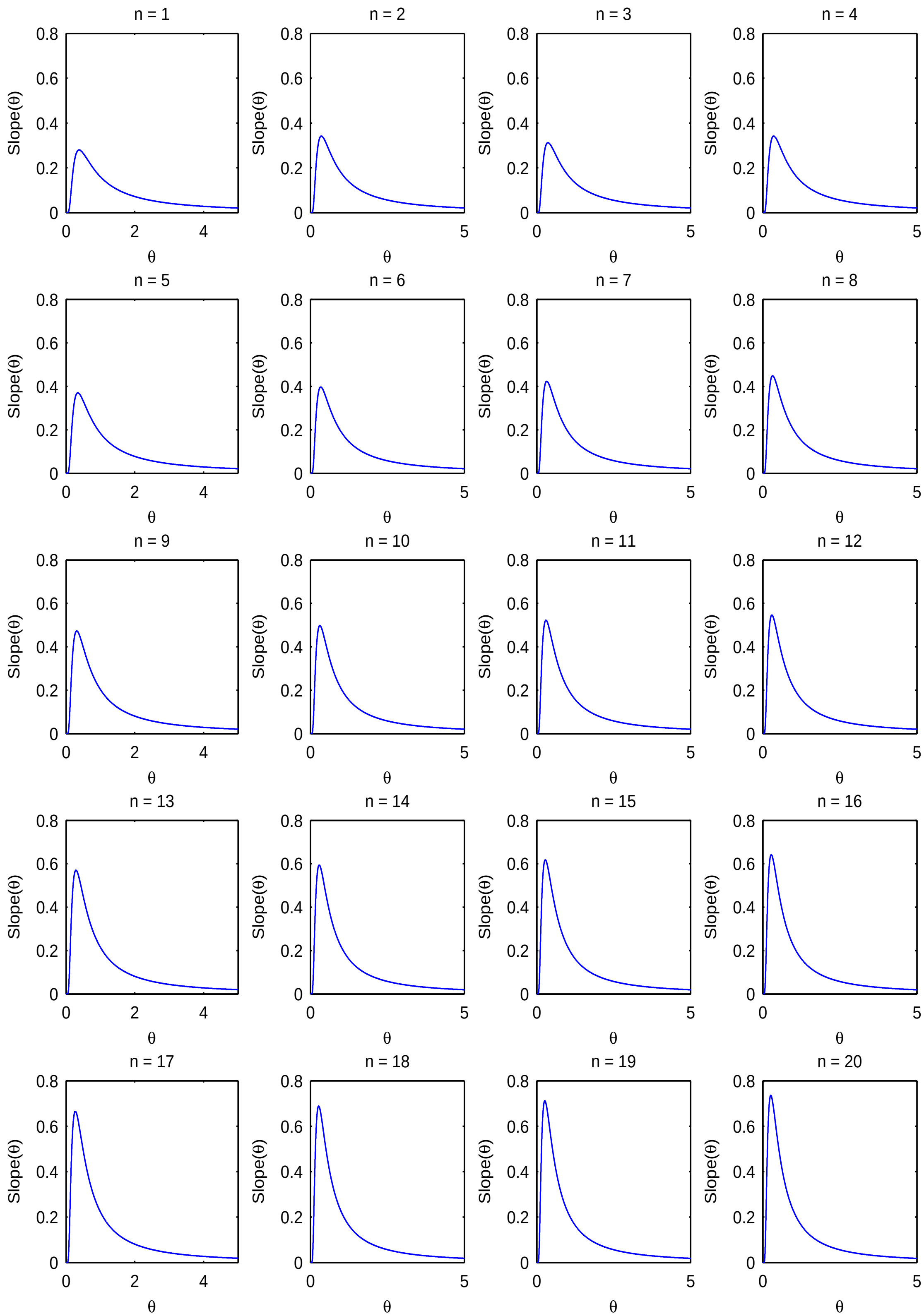

Fig. 25. $(\theta, Slope(\theta))$ for $K = 0.1$, $\theta_0 = 0.5$

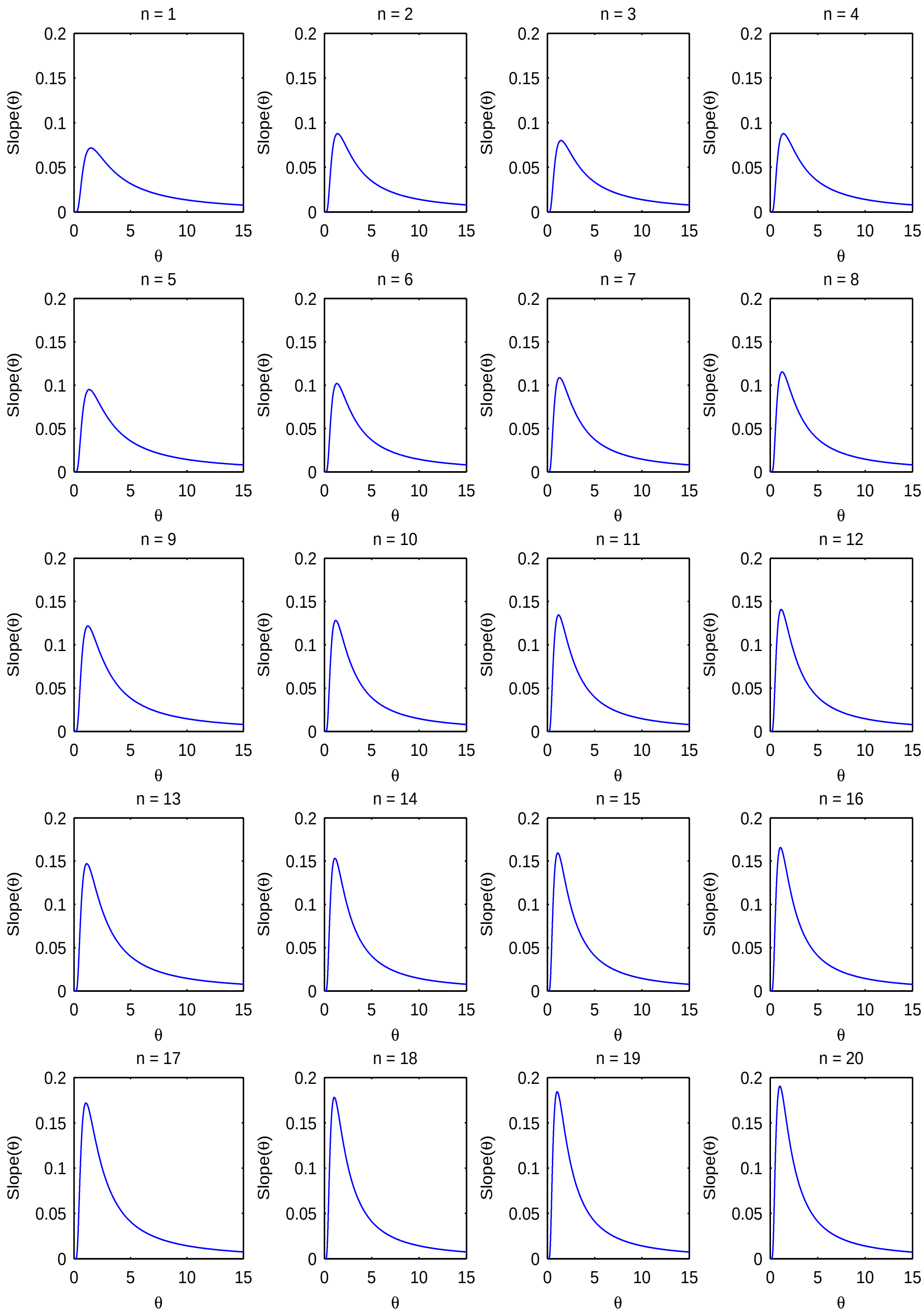

Fig. 26. $\big(\theta, Slope(\theta)\big)$ for $K = 0.1$, $\theta_0 = 2$

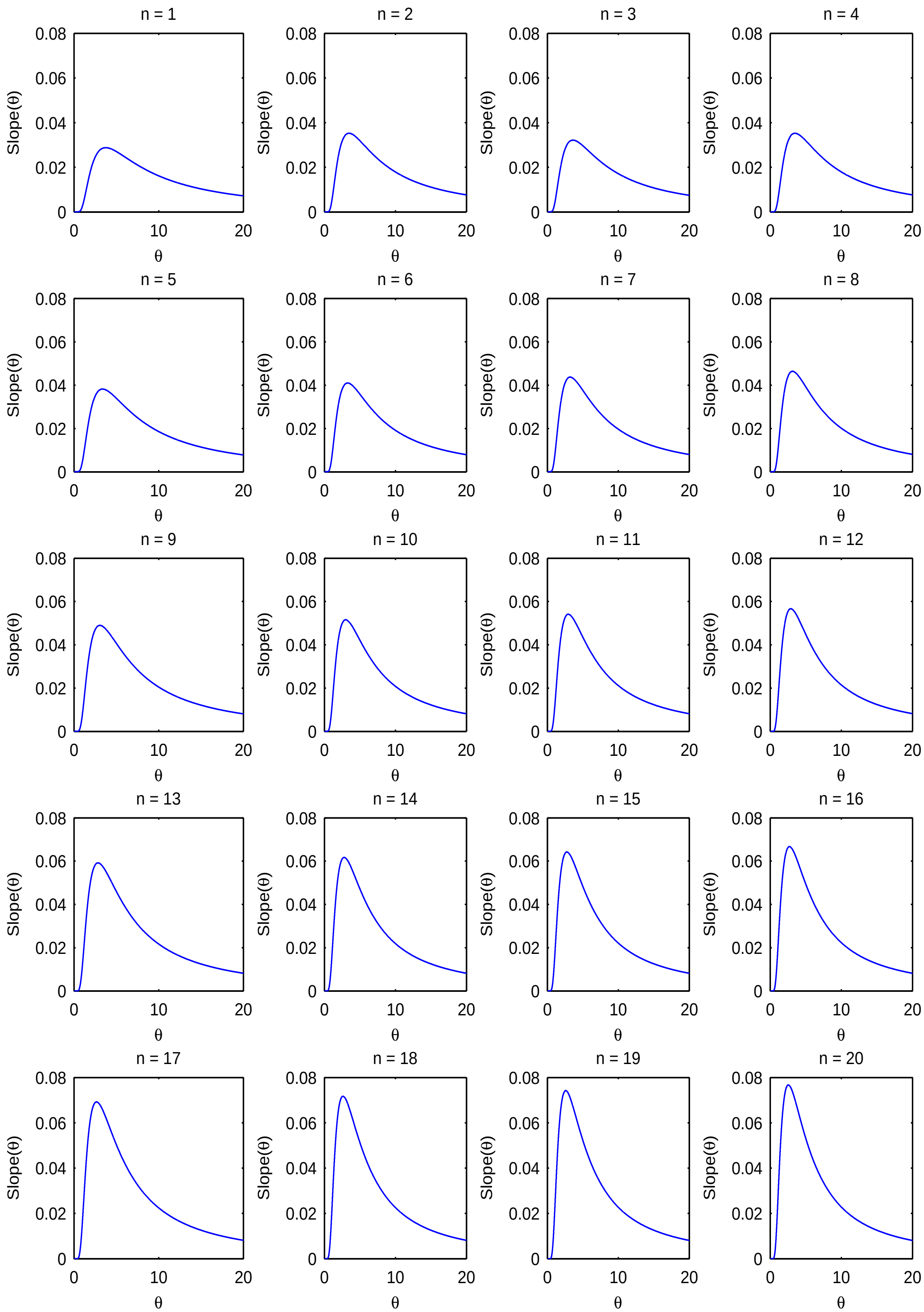

Fig. 27. $(\theta, Slope(\theta))$ for $K = 0.1$, $\theta_0 = 5$

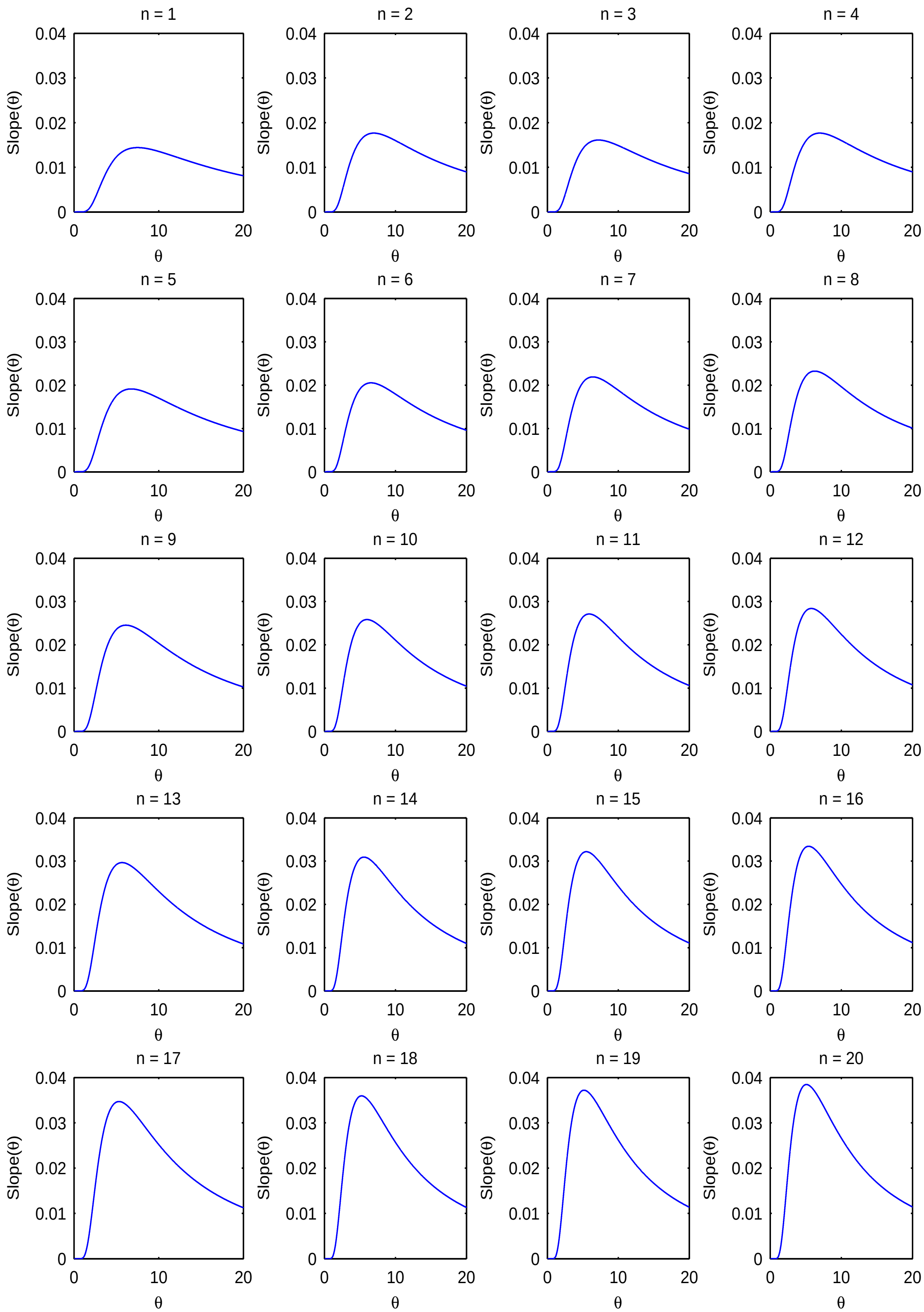

Fig. 28. $\big(\theta, Slope(\theta)\big)$ for $K = 0.1$, $\theta_0 = 10$

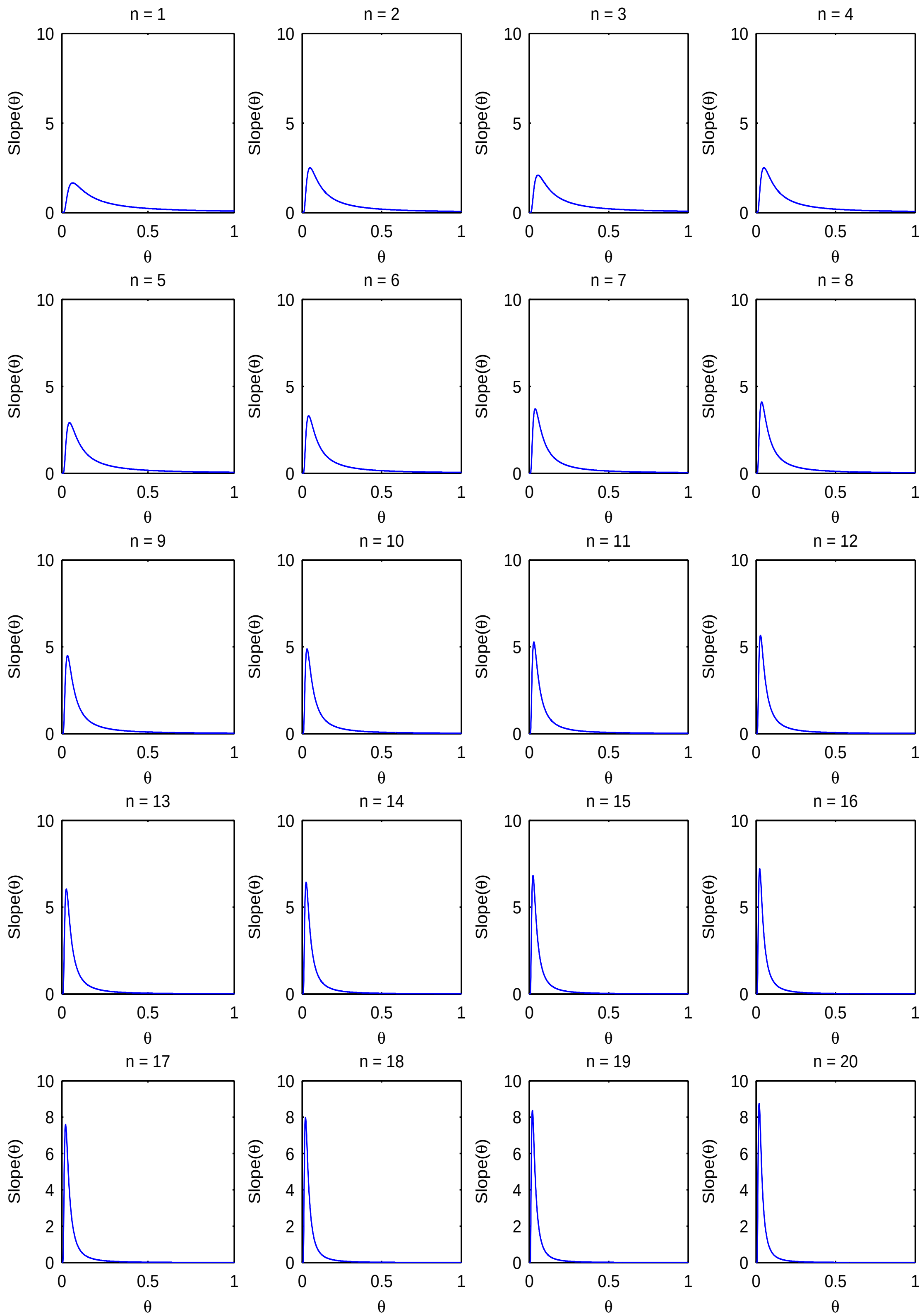

Fig. 29. $\big(\theta, Slope(\theta)\big)$ for $K = 1$, $\theta_0 = 0.1$

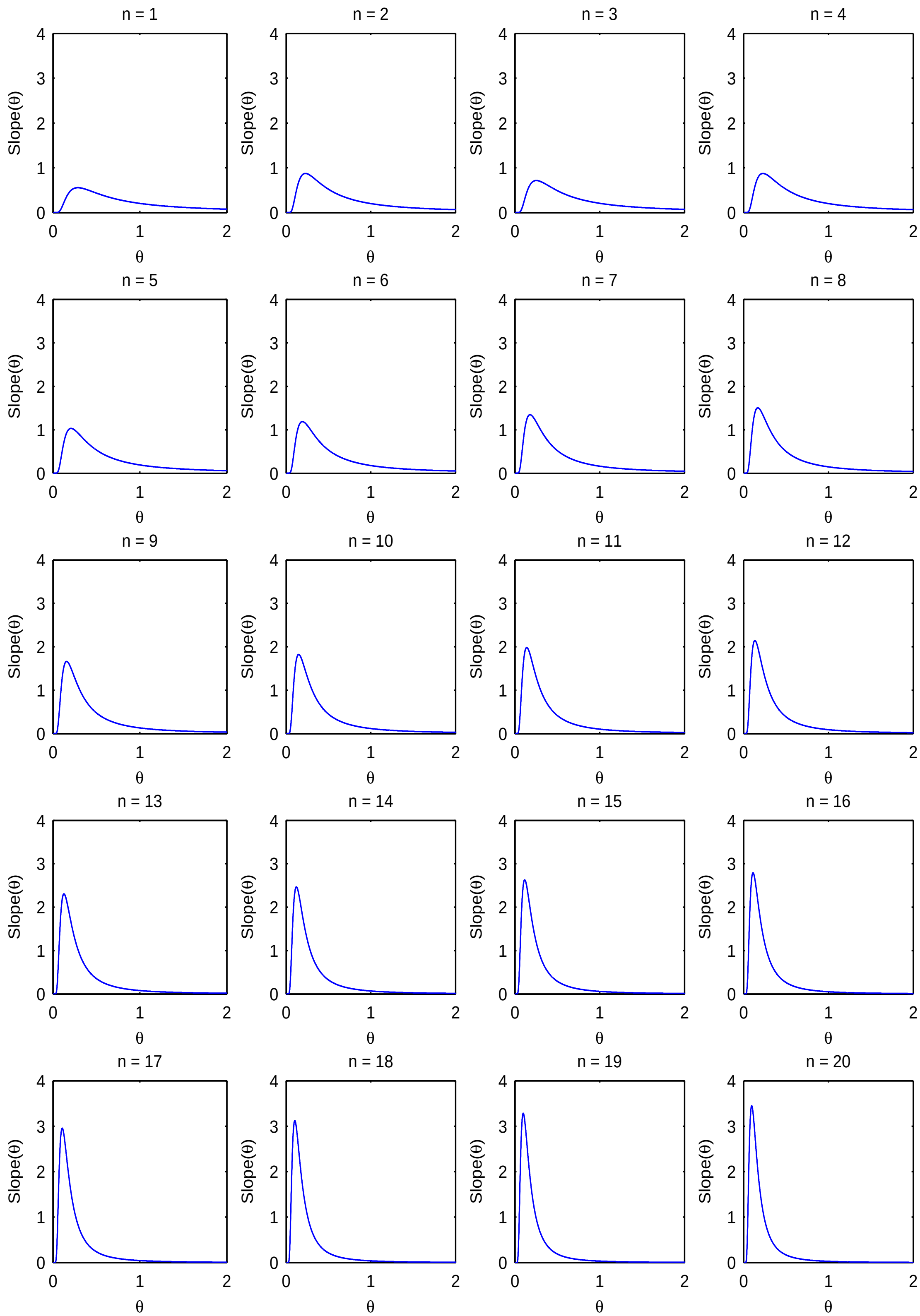

Fig. 30. $\big(\theta, Slope(\theta)\big)$ for $K = 1$, $\theta_0 = 0.5$

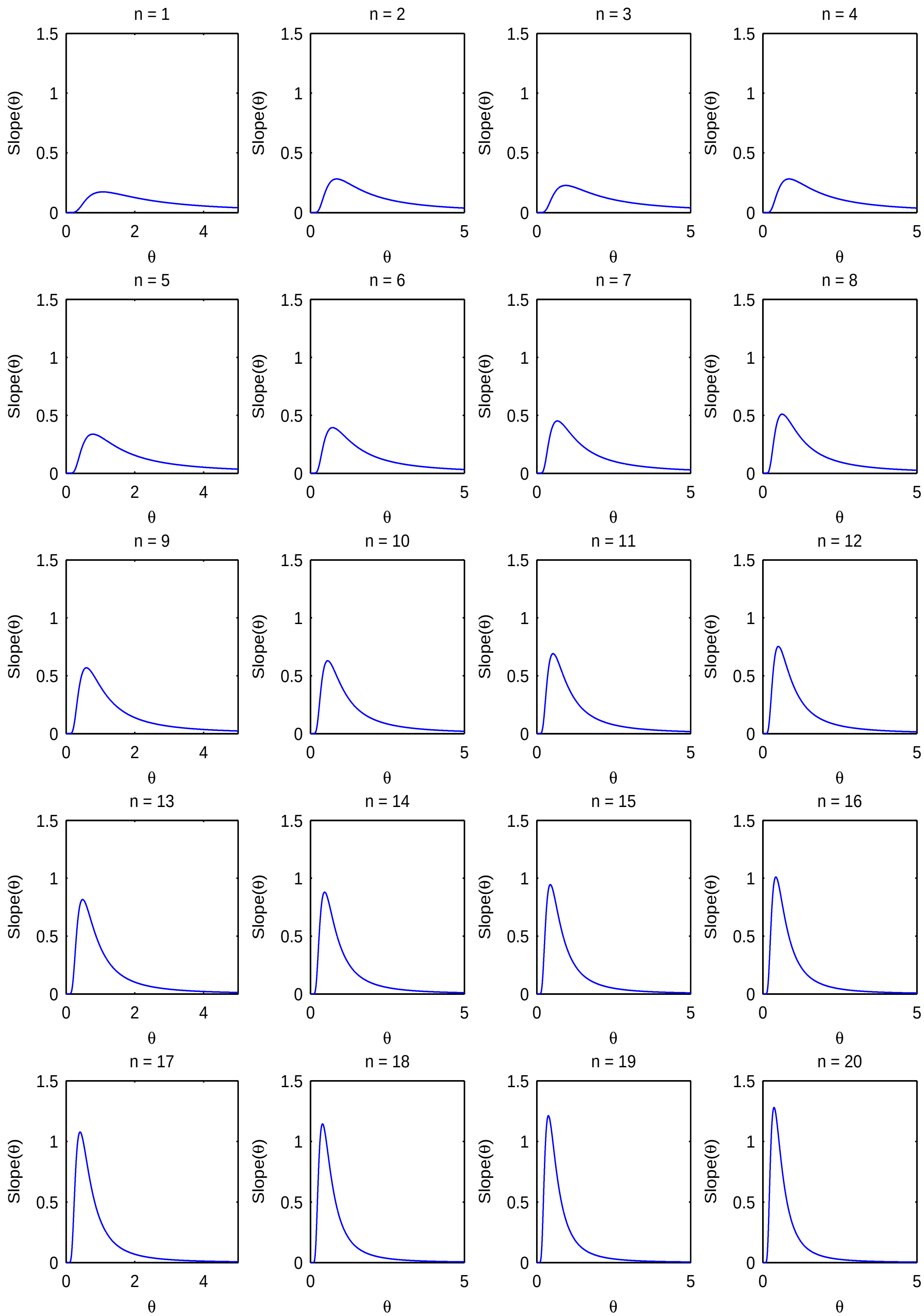

Fig. 31. $\left(\theta, Slope(\theta)\right)$ for $K = 1$, $\theta_0 = 2$

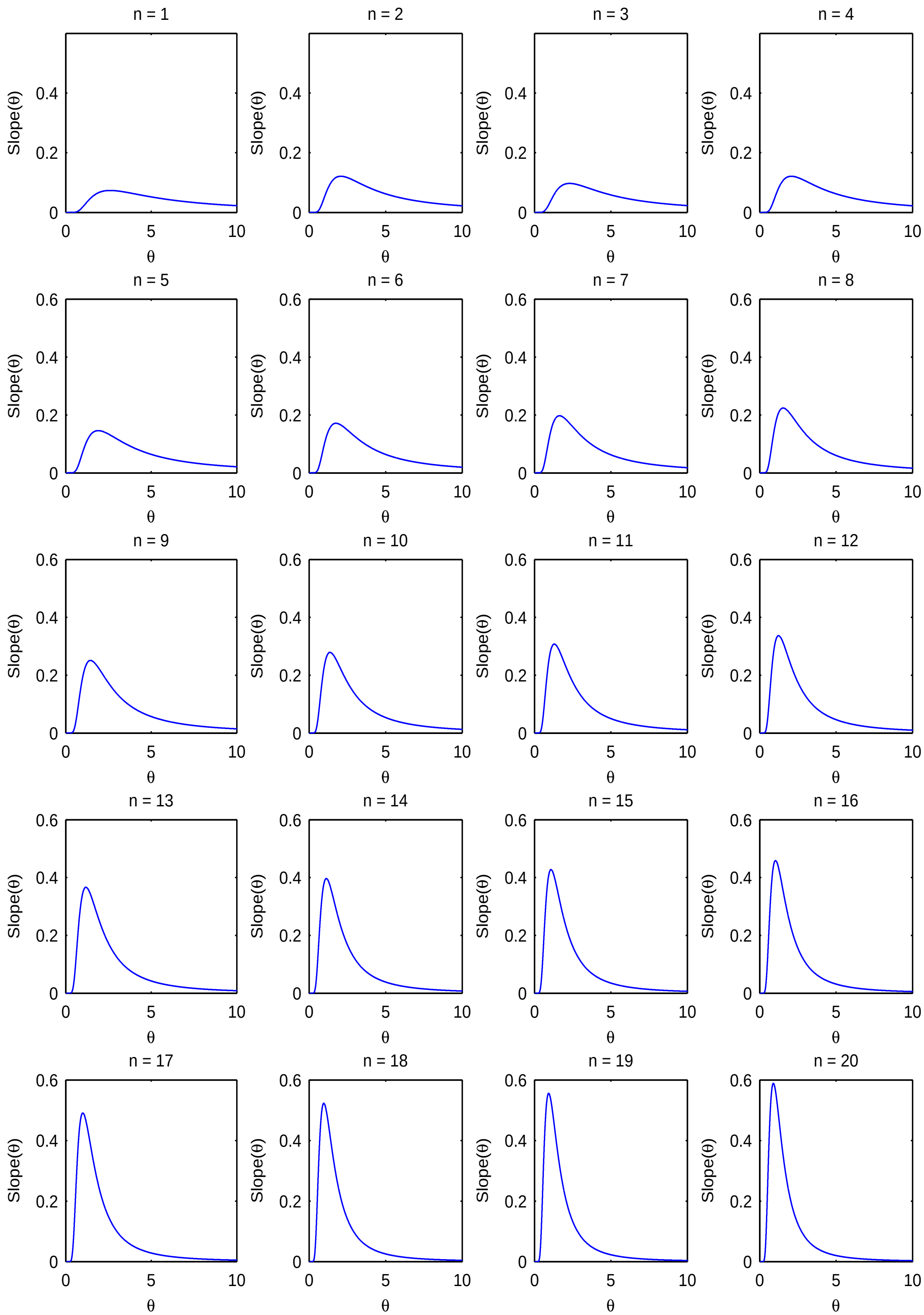

Fig. 32. $\big(\theta, Slope(\theta)\big)$ for $K = 1$, $\theta_0 = 5$

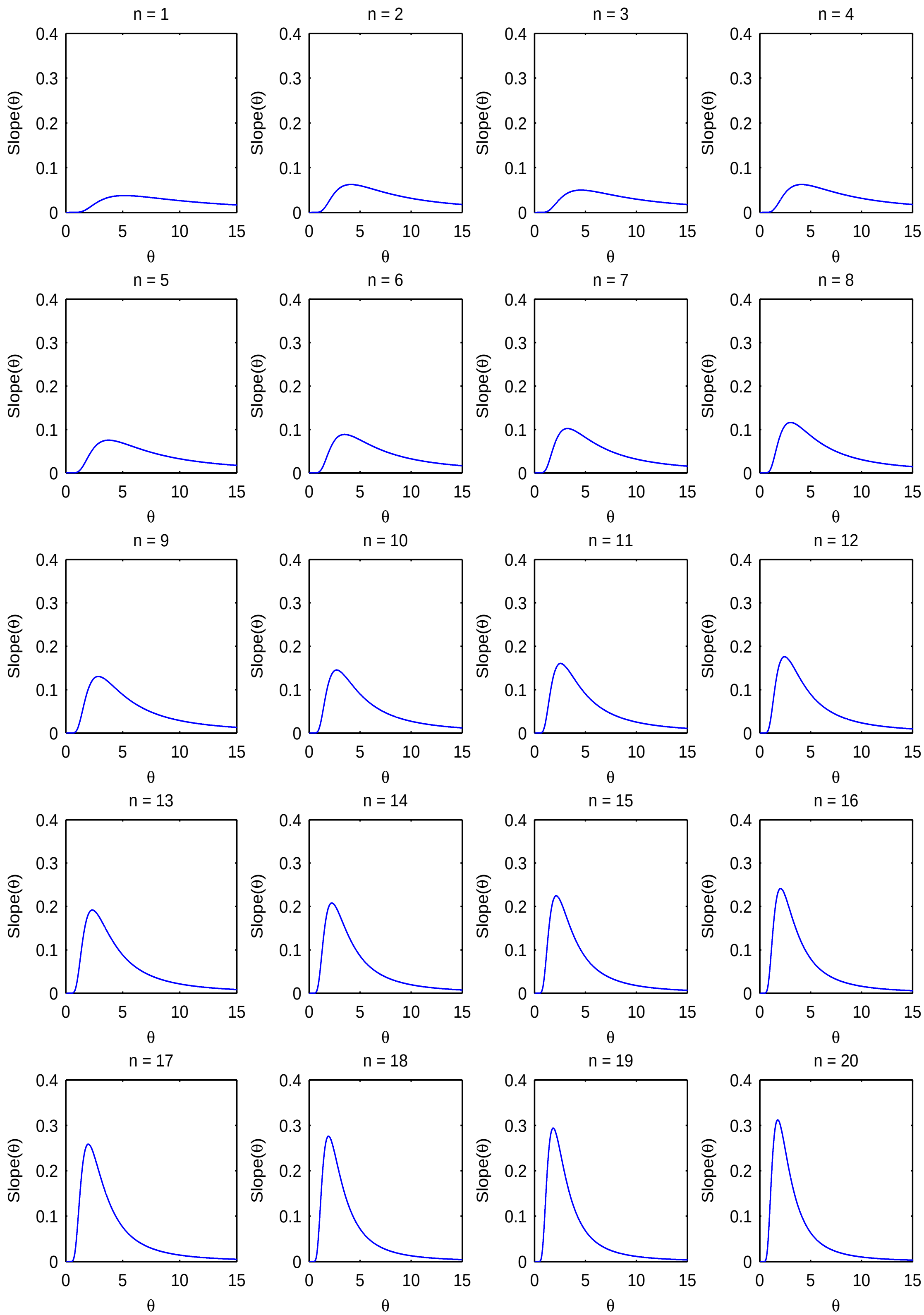

Fig. 33. $\big(\theta, Slope(\theta)\big)$ for $K = 1$, $\theta_0 = 10$

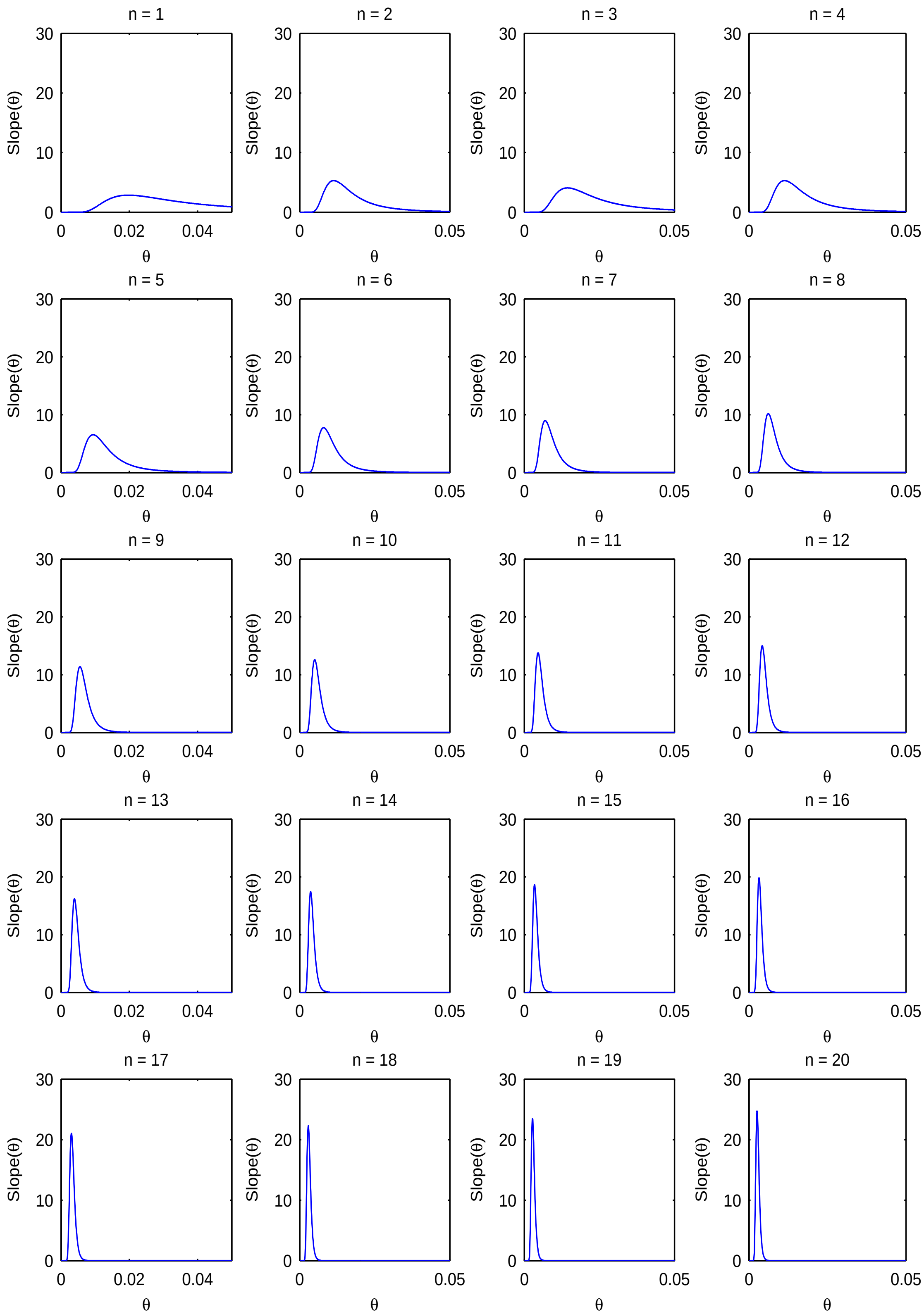

Fig. 34. $\big(\theta, Slope(\theta)\big)$ for $K = 10$, $\theta_0 = 0.1$

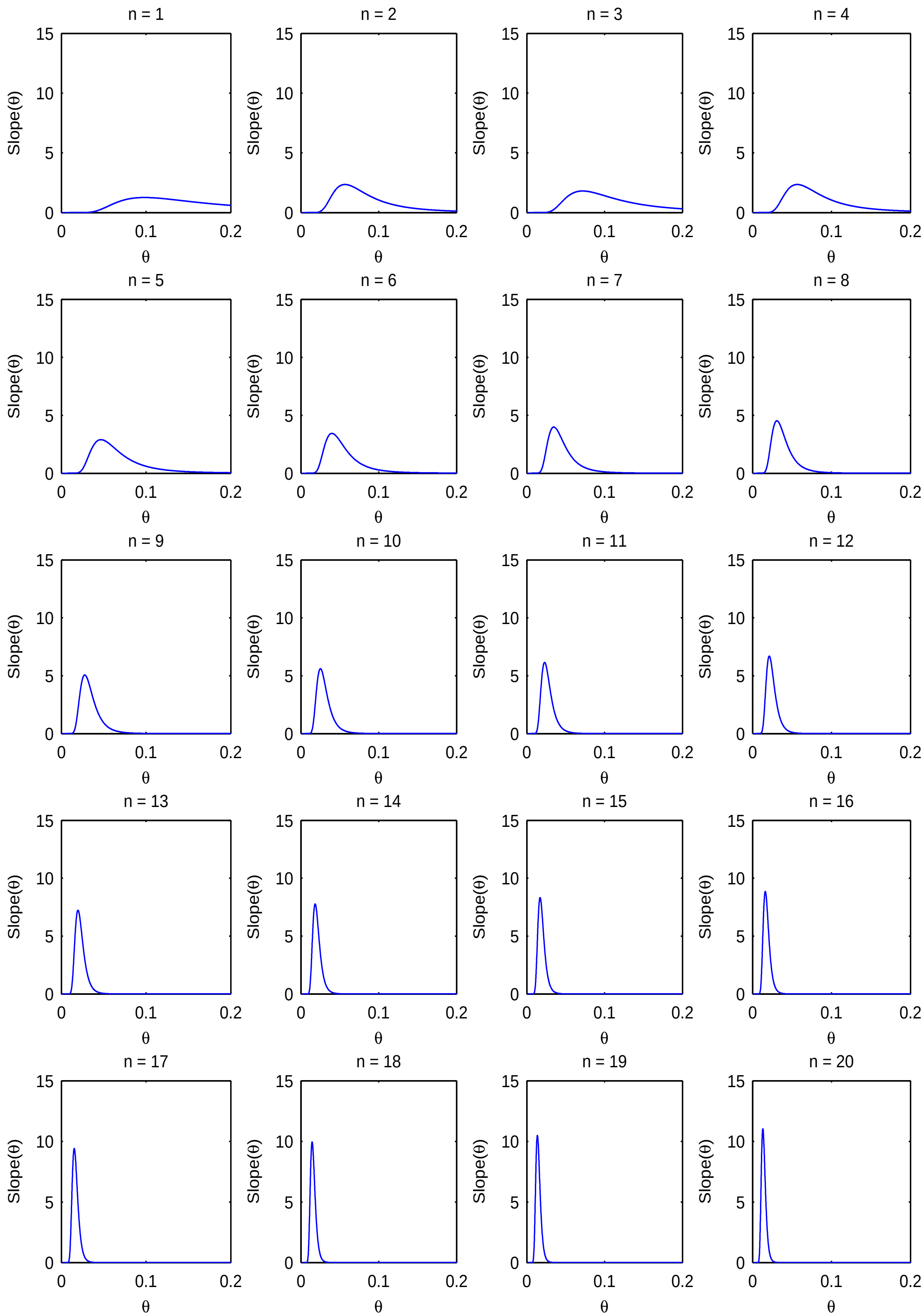

Fig. 35. $\big(\theta, Slope(\theta)\big)$ for $K = 10$, $\theta_0 = 0.5$

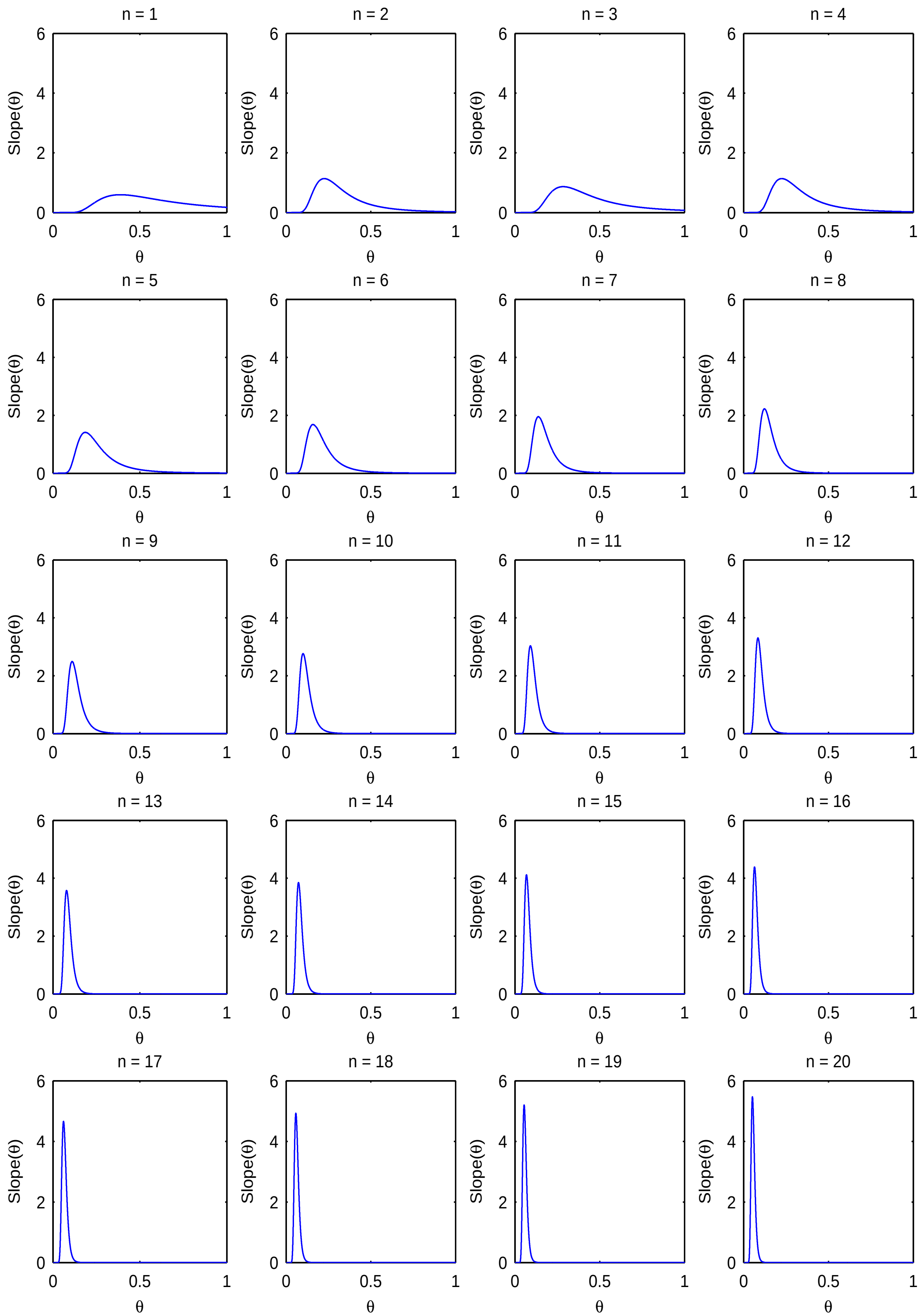

Fig. 36. $\big(\theta, Slope(\theta)\big)$ for $K = 10$, $\theta_0 = 2$

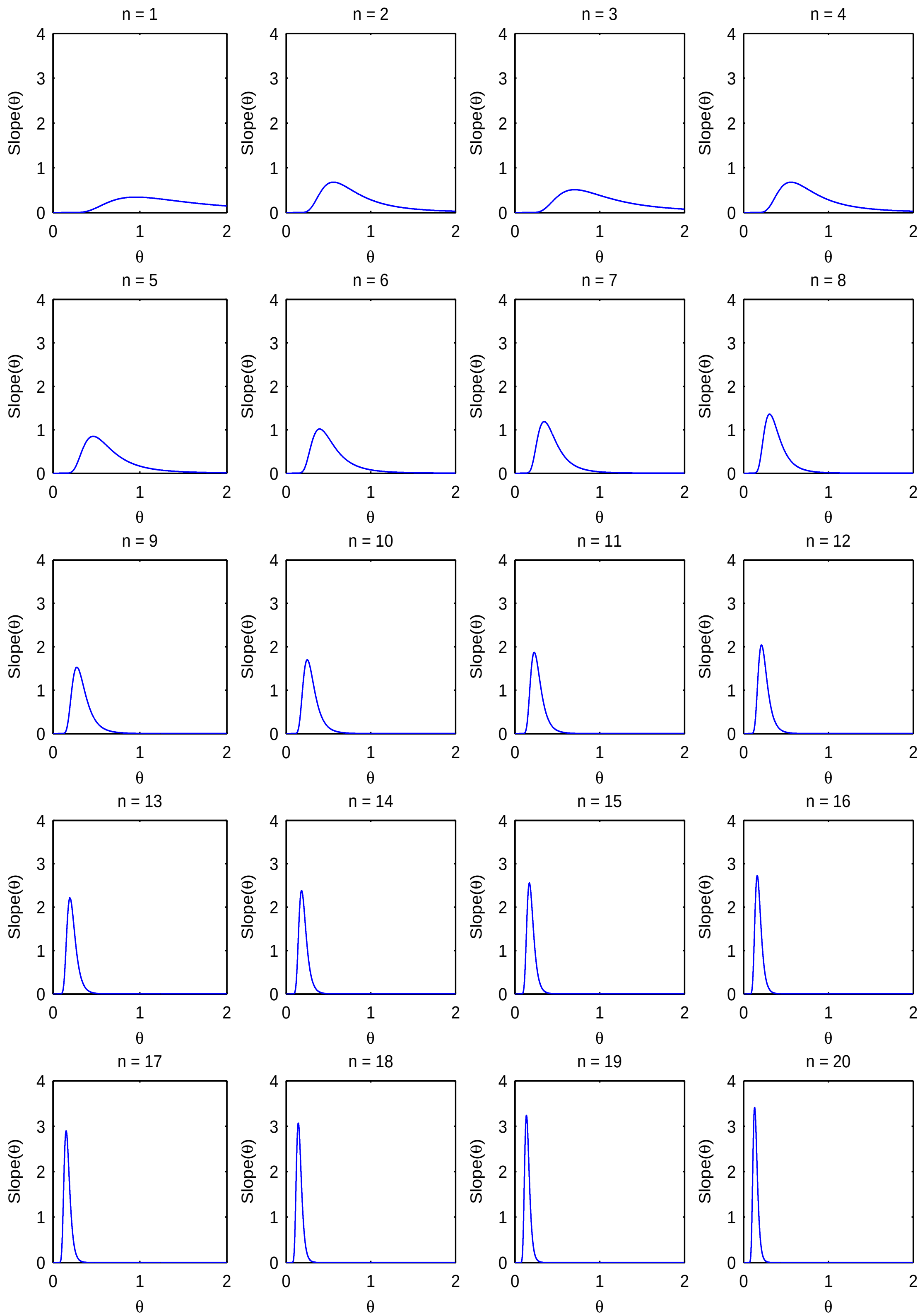

Fig. 37. $(\theta, Slope(\theta))$ for $K = 10$, $\theta_0 = 5$

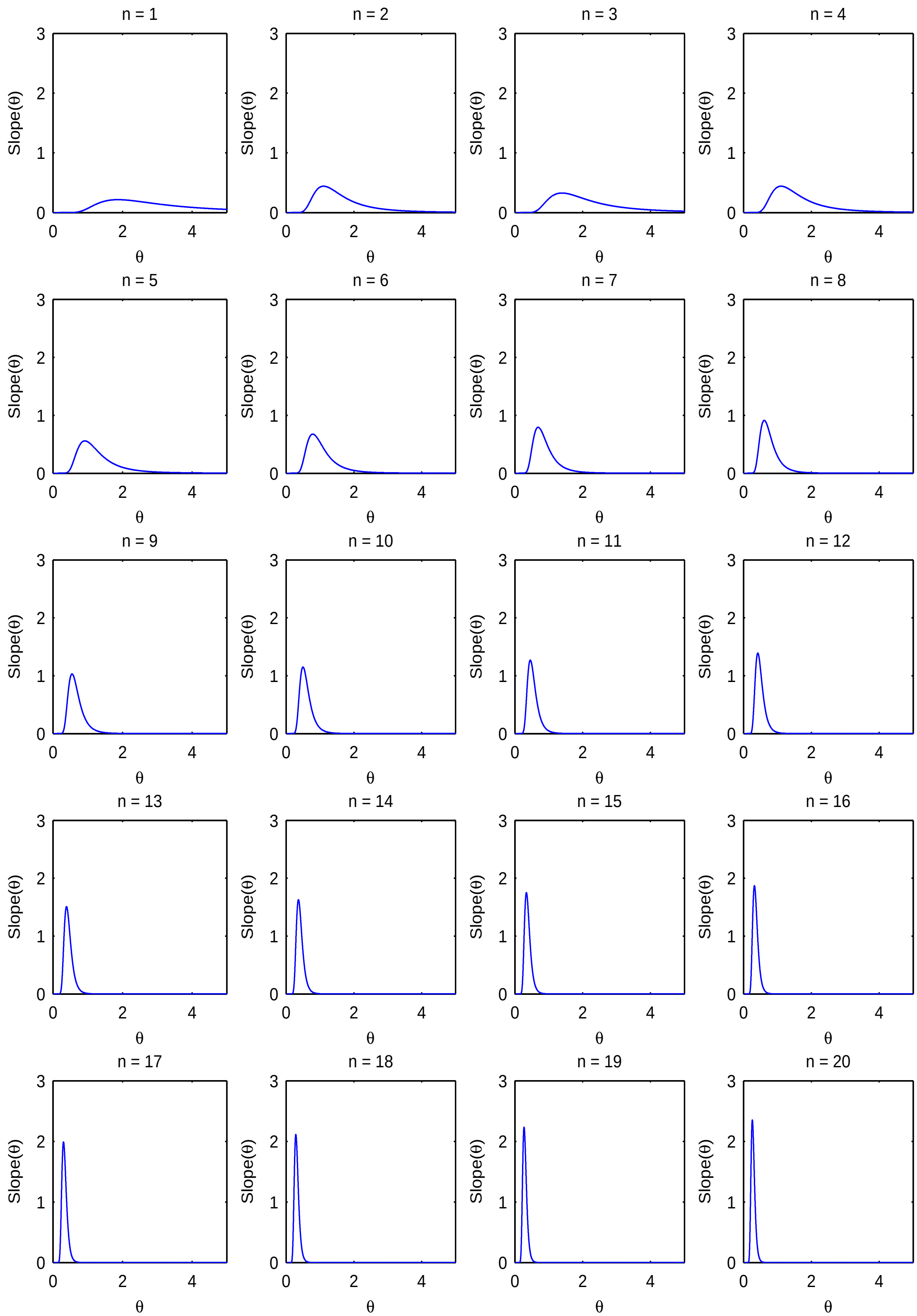

Fig. 38. $\big(\theta, Slope(\theta)\big)$ for $K = 10$, $\theta_0 = 10$

APPENDIX B

SLOPE LIKELIHOOD CURVES

Likelihood of $Slope(\theta)$ for $n = 1$ - 20, $K = \{0.1, 1, 4\}$, and $\theta_0 = \{0.1, 0.5, 2, 5, 10\}$ are given in Figs. 39 - 53.

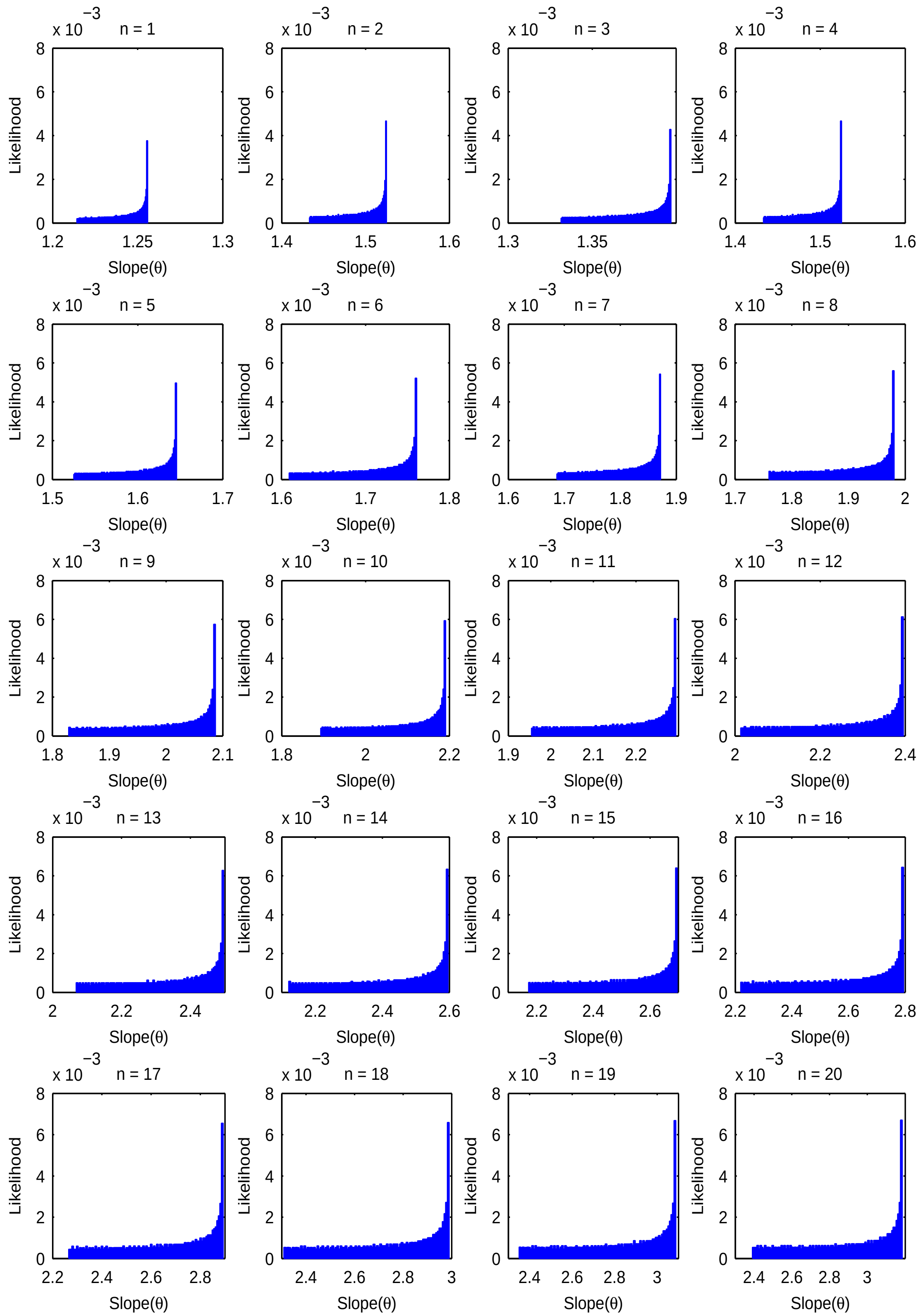

Fig. 39. SL curve for $K = 0.1$, $\theta_0 = 0.1$

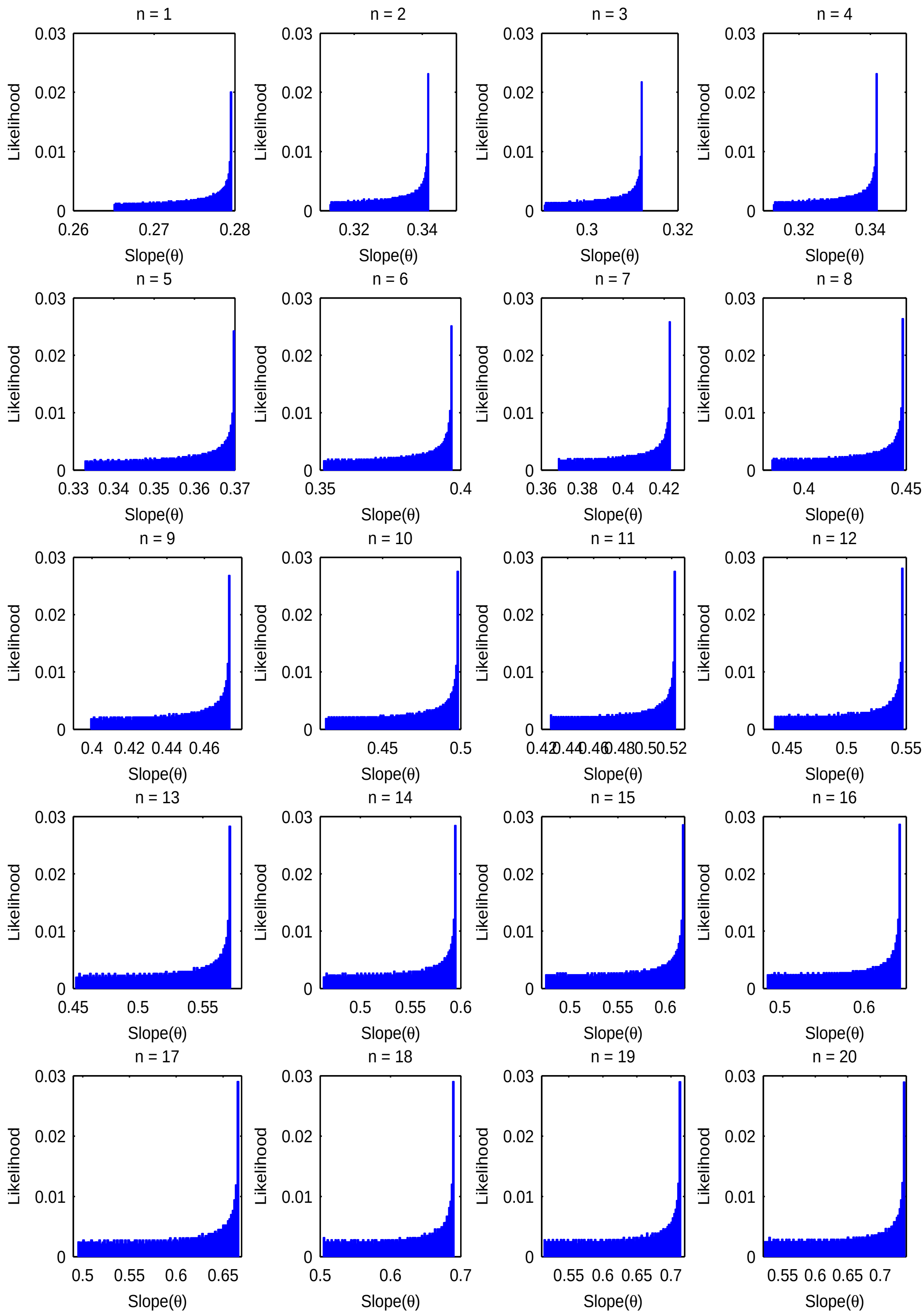

Fig. 40. SL curve for $K = 0.1$, $\theta_0 = 0.5$

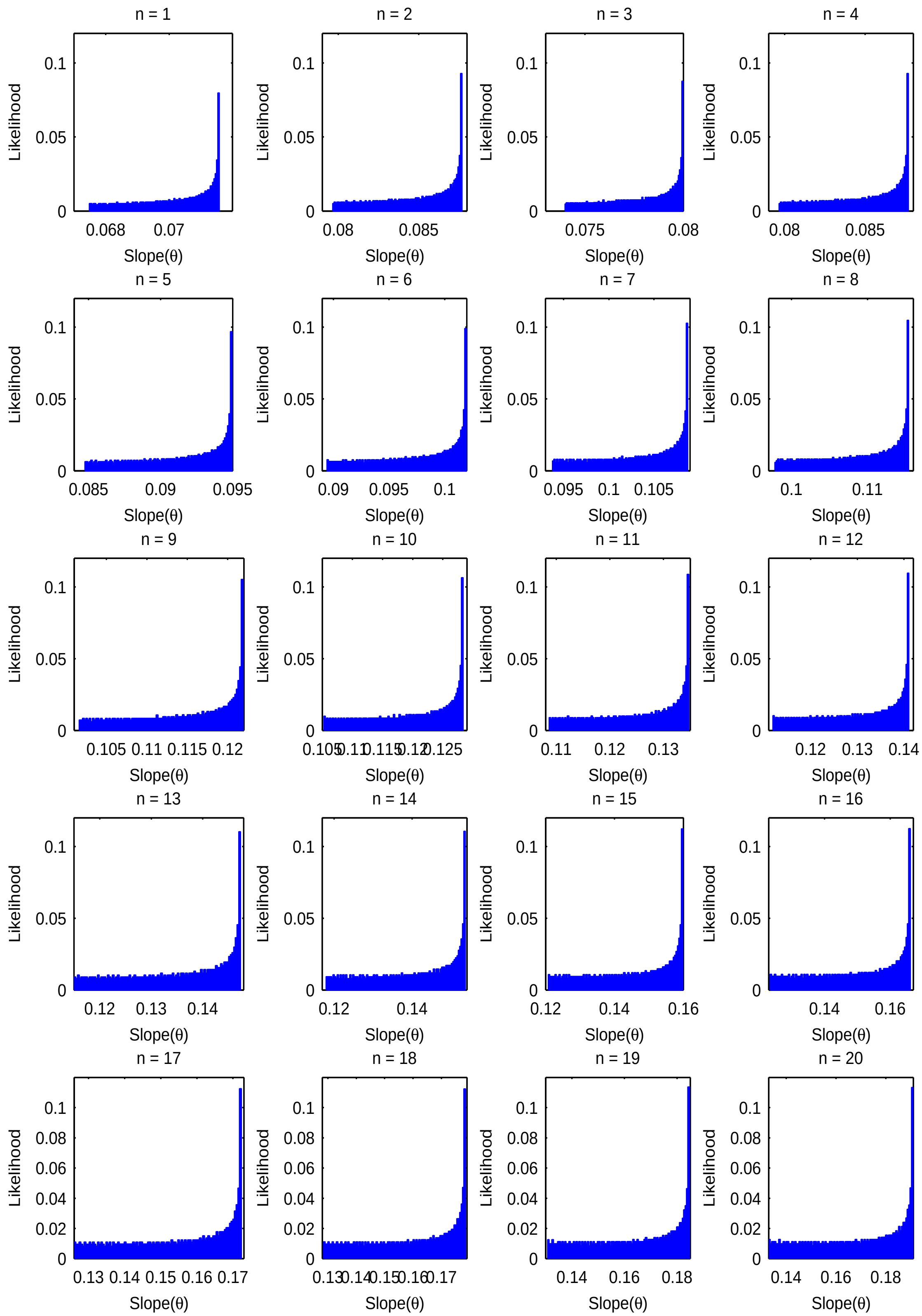

Fig. 41. SL curve for $K = 0.1$, $\theta_0 = 2$

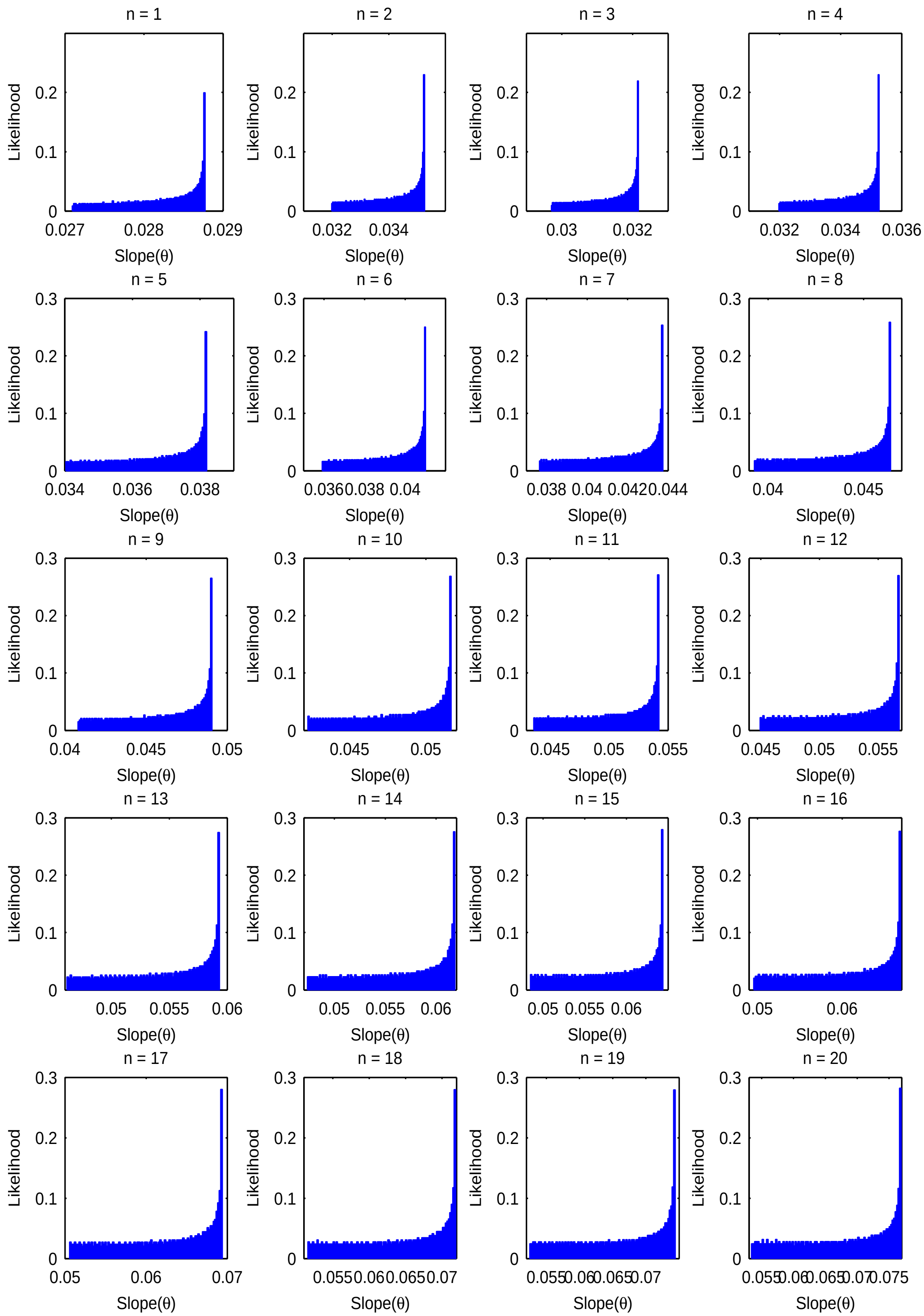

Fig. 42. SL curve for $K = 0.1$, $\theta_0 = 5$

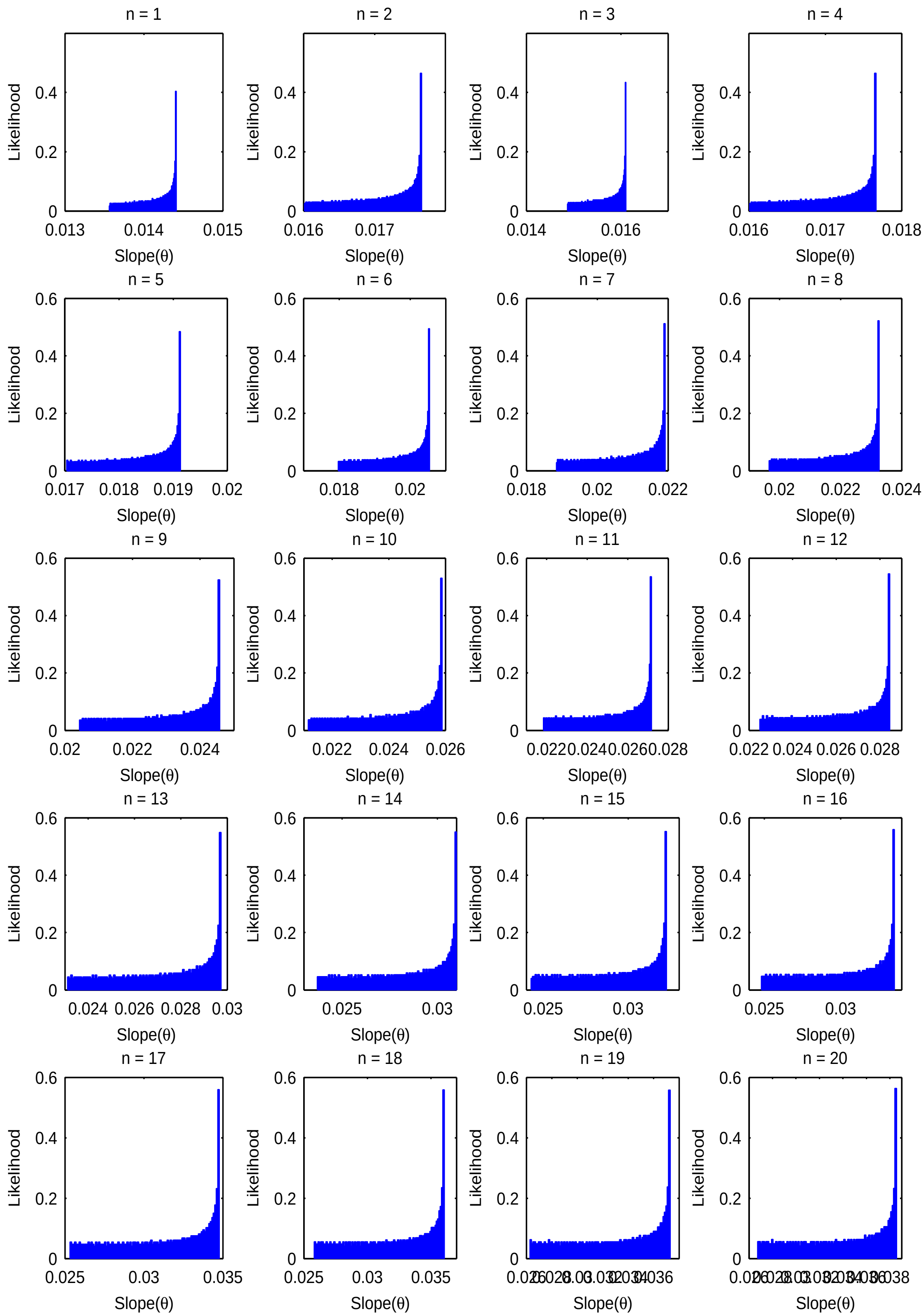

Fig. 43. SL curve for $K = 0.1$, $\theta_0 = 10$

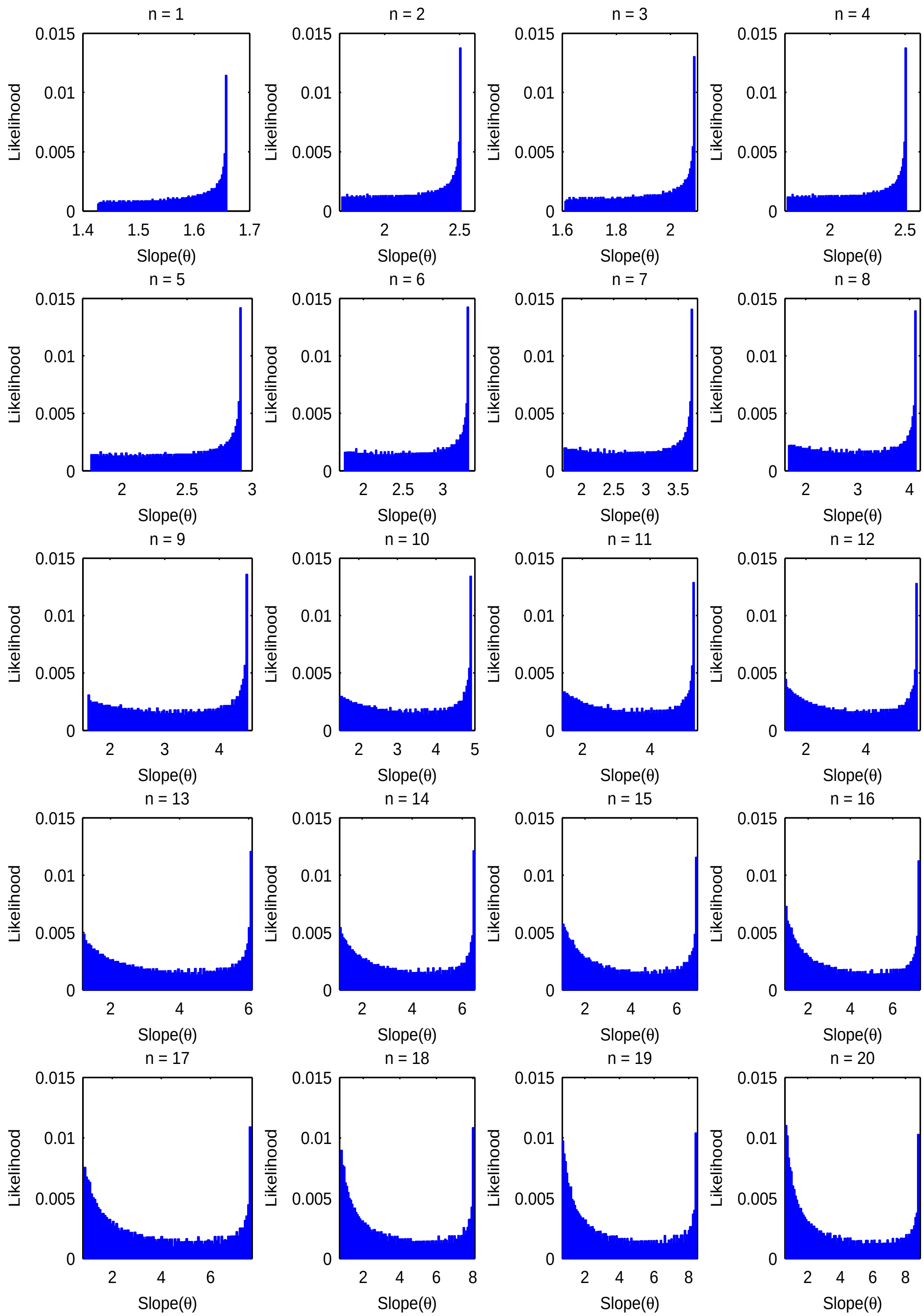

Fig. 44. SL curve for $K = 1$, $\theta_0 = 0.1$

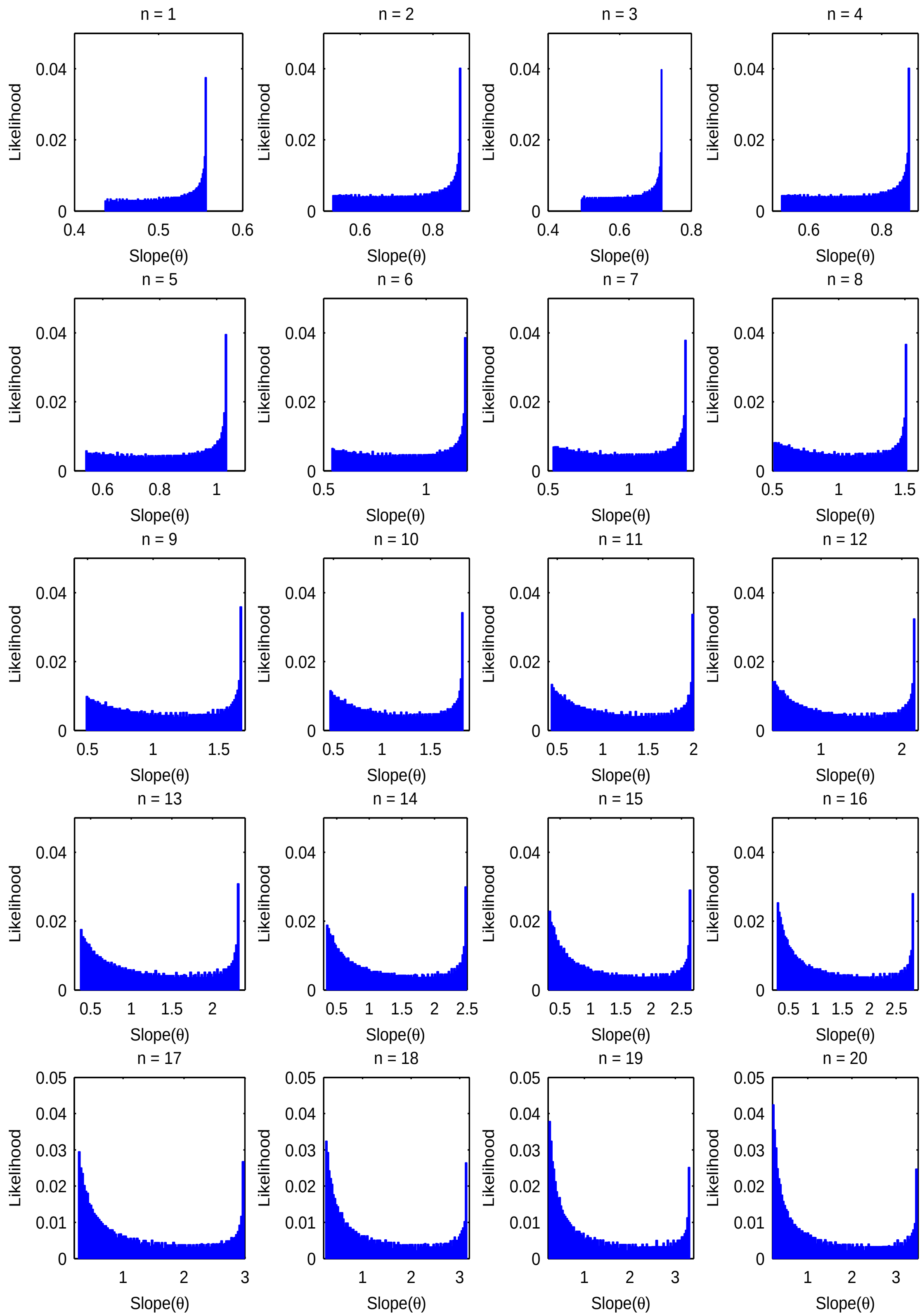

Fig. 45. SL curve for $K = 1$, $\theta_0 = 0.5$

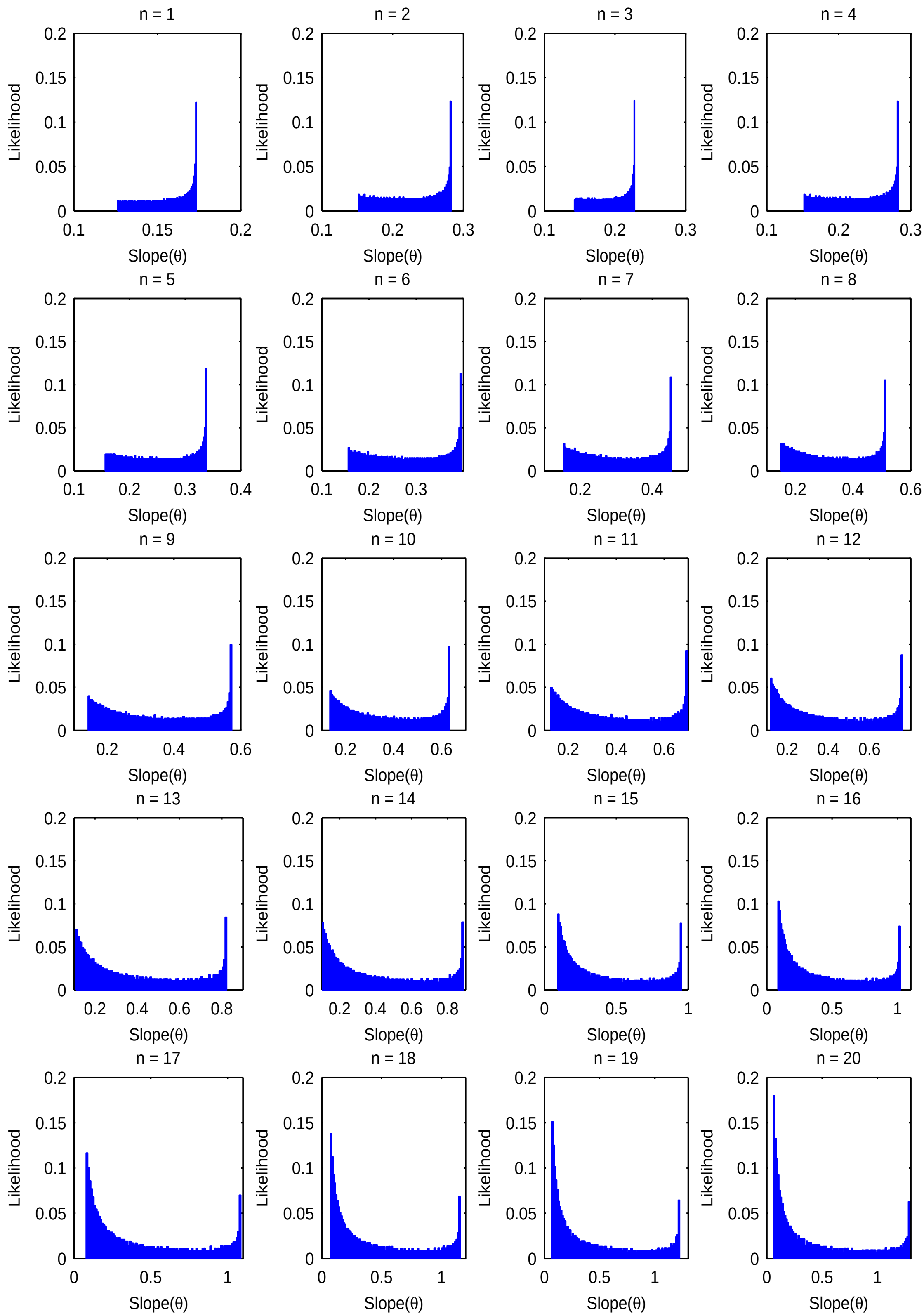

Fig. 46. SL curve for $K = 1$, $\theta_0 = 2$

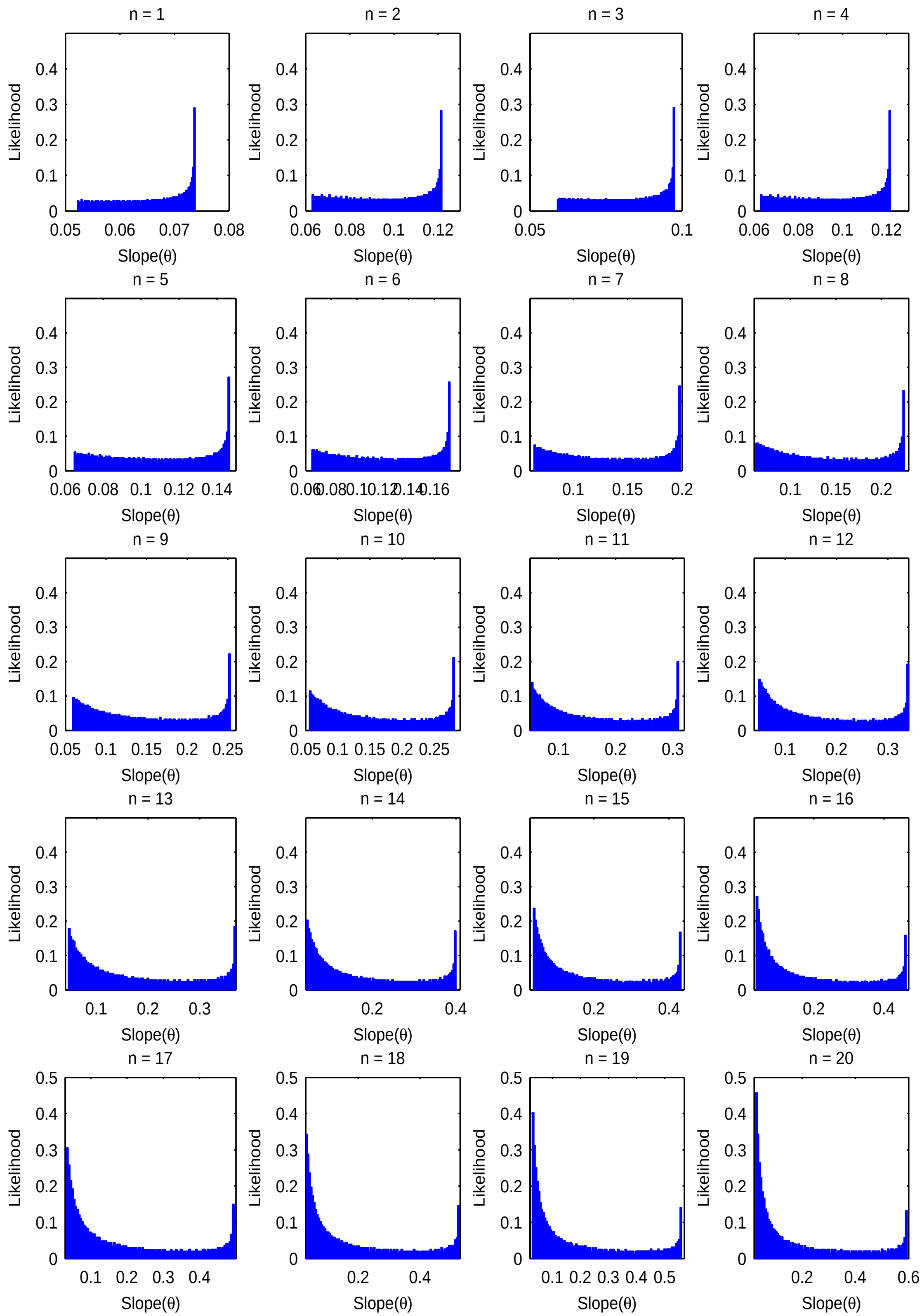

Fig. 47. SL curve for $K = 1$, $\theta_0 = 5$

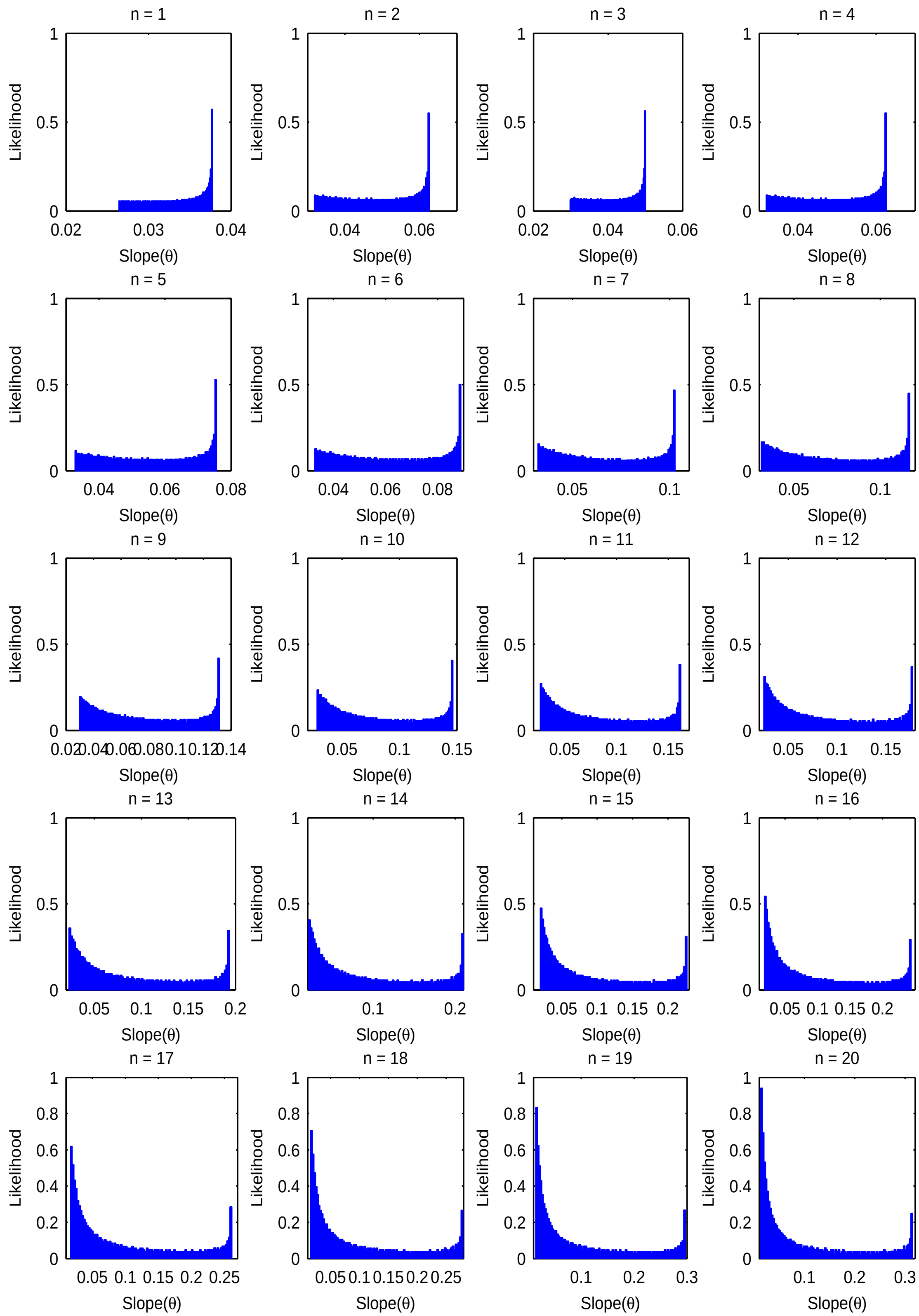

Fig. 48. SL curve for $K = 1$, $\theta_0 = 10$

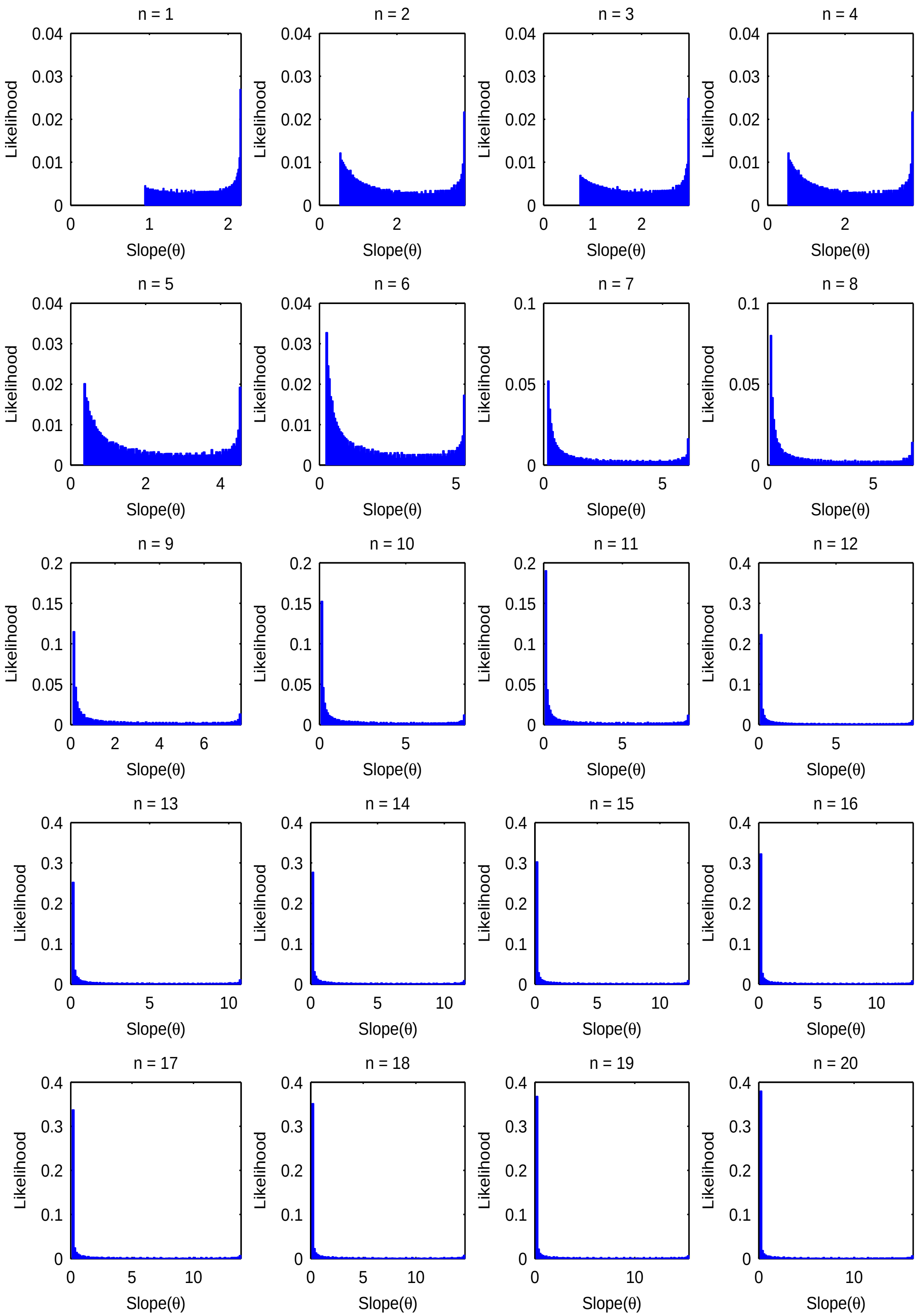

Fig. 49. SL curve for $K = 4$, $\theta_0 = 0.1$

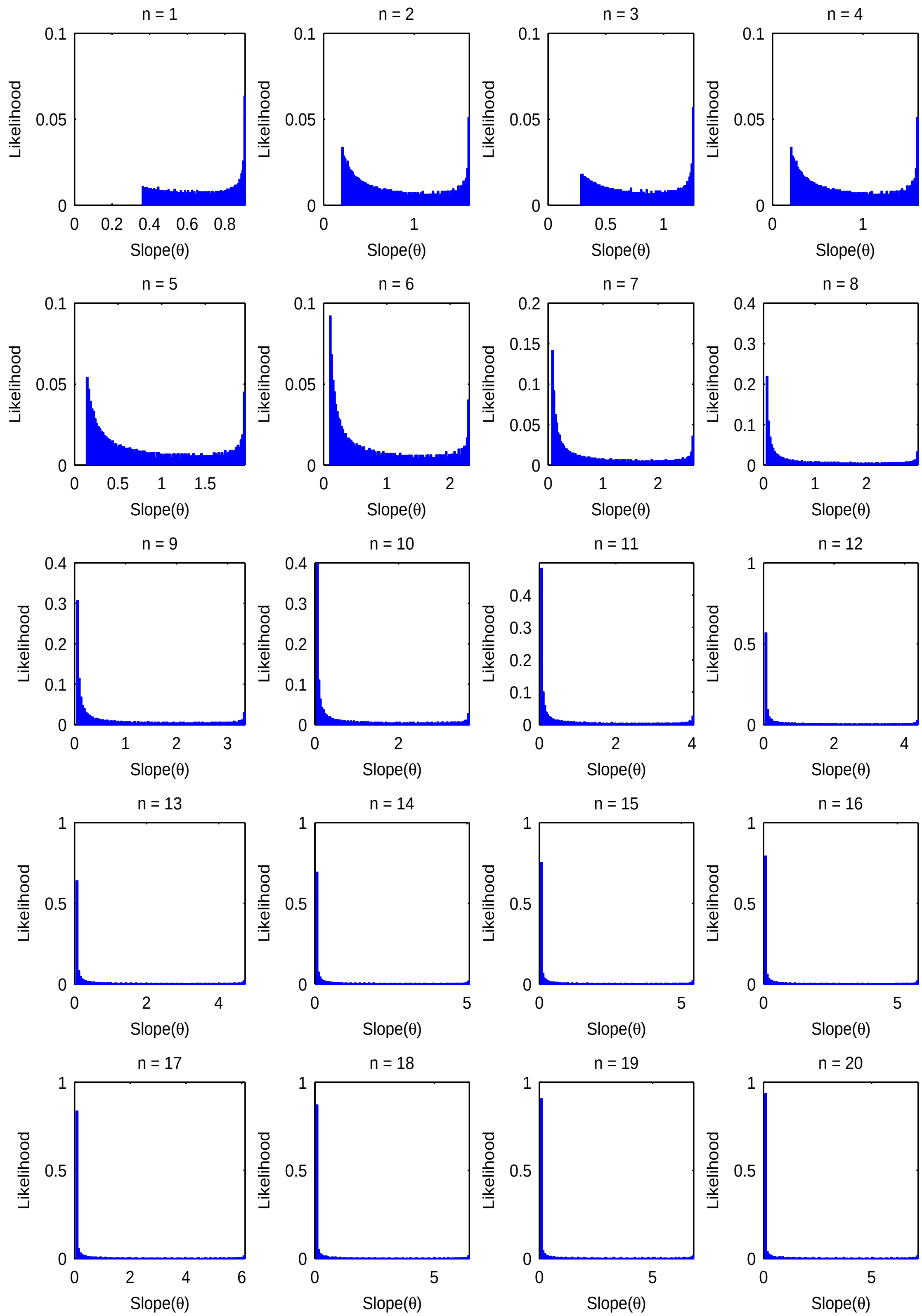

Fig. 50. SL curve for $K = 4$, $\theta_0 = 0.5$

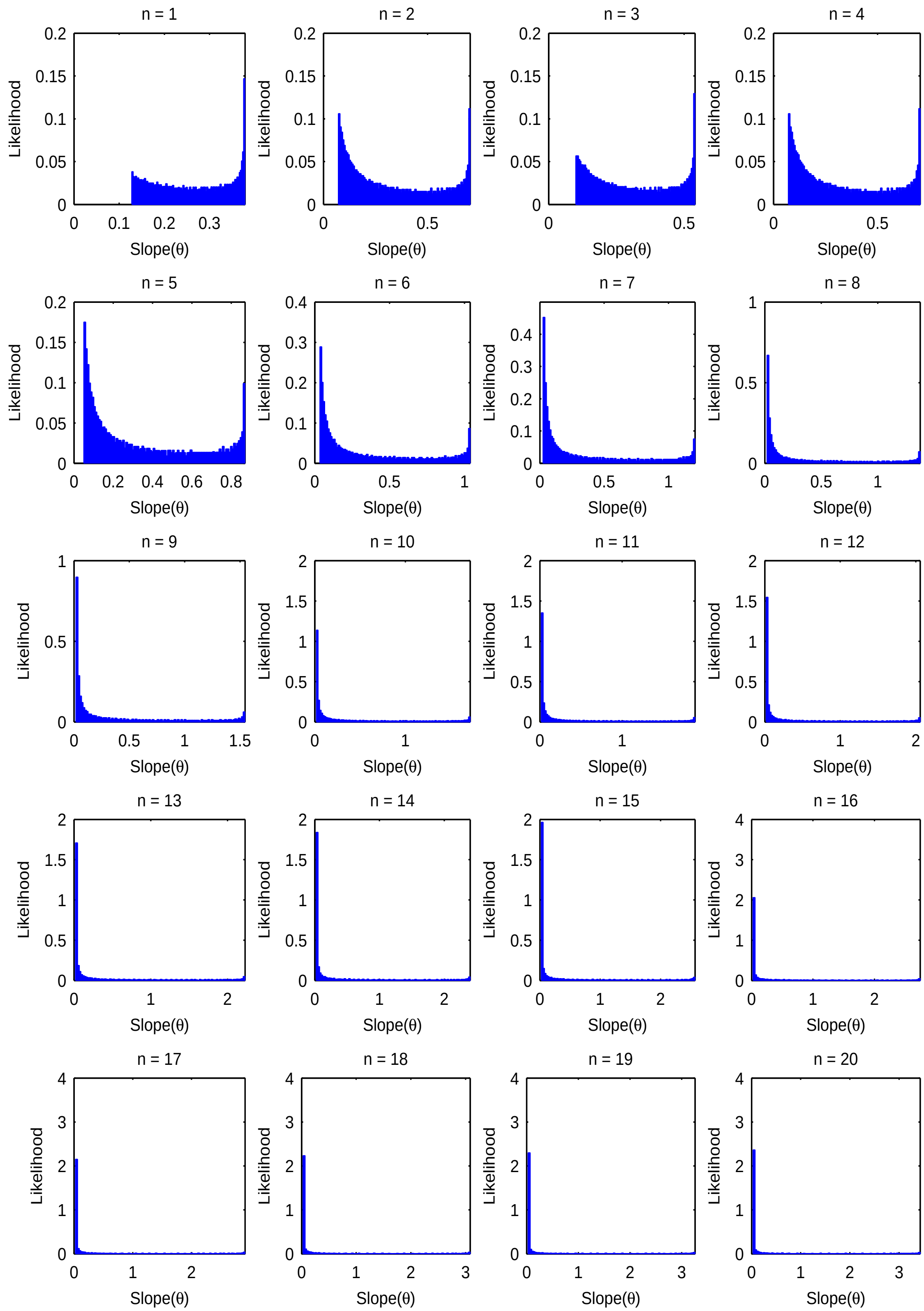

Fig. 51. SL curve for $K = 4$, $\theta_0 = 2$

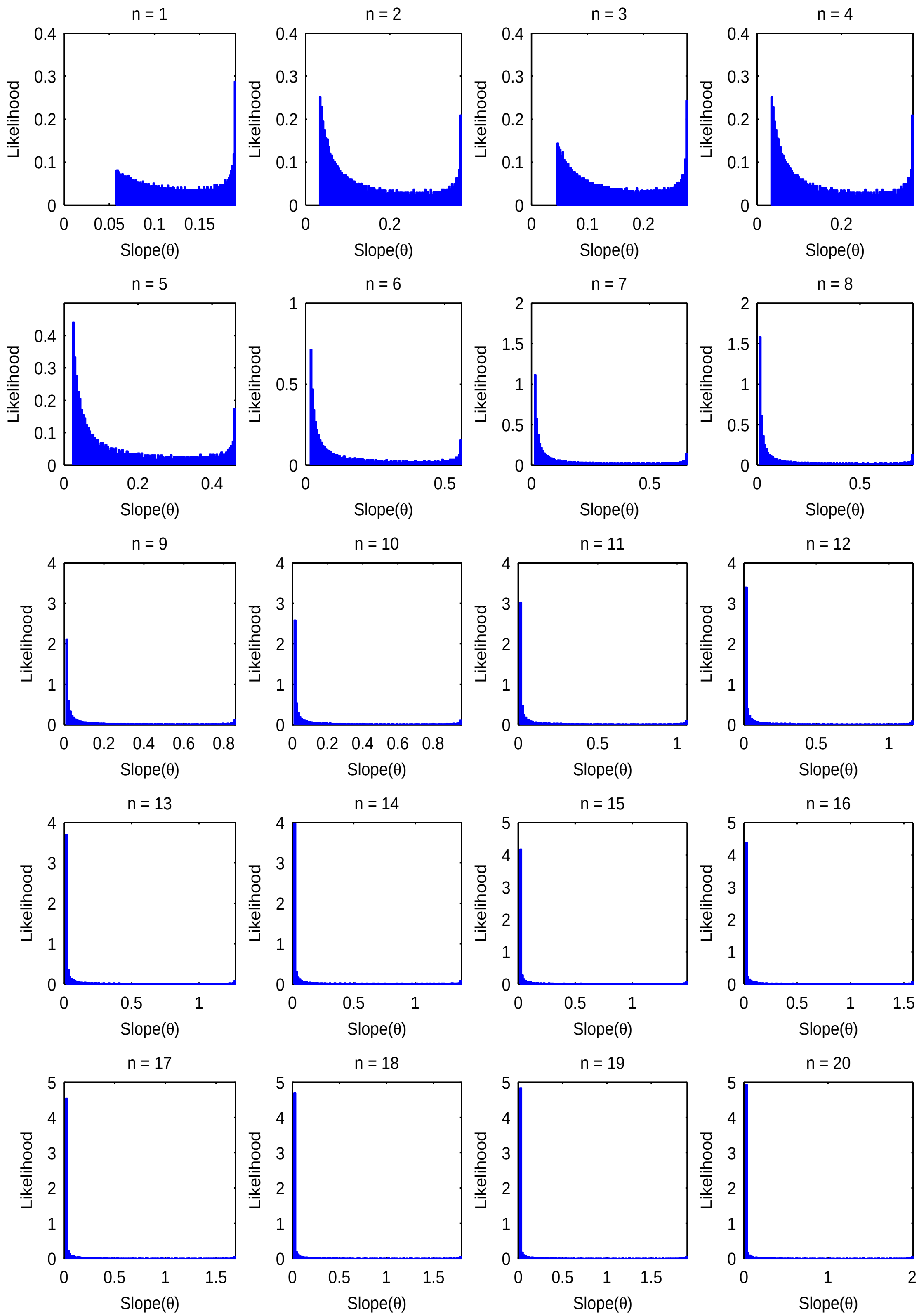

Fig. 52. SL curve for $K = 4$, $\theta_0 = 5$

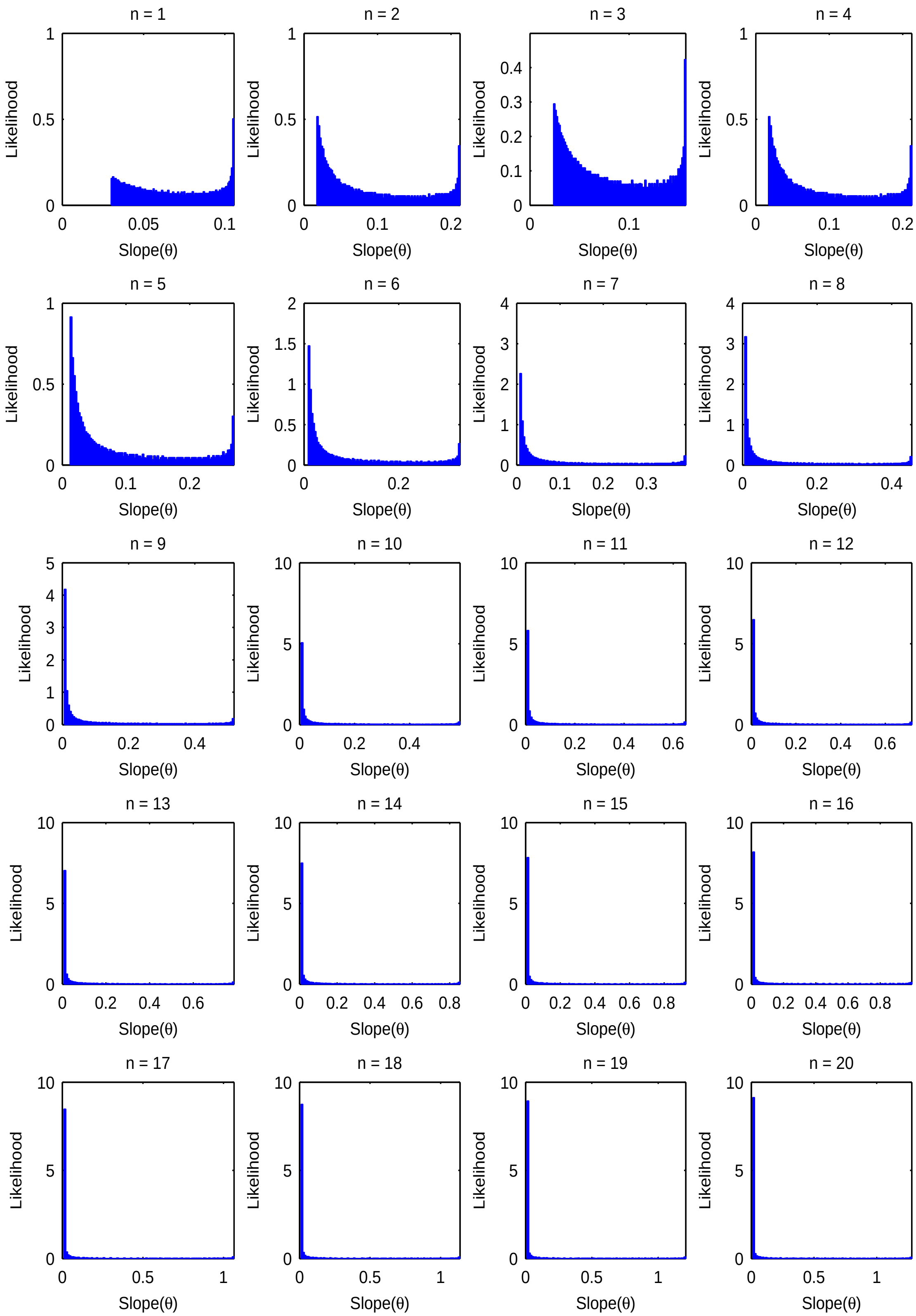

Fig. 53. SL curve for $K = 4$, $\theta_0 = 10$

APPENDIX C

CONFIDENCE PROCEDURE RESULTS

The confidence (%C) and weighted confidence (%WC) computational results for $n = 1$ - 20, $K = \{0.1, 1, 10\}$, and $\theta_0 = \{0.1, 0.5, 2, 5, 10\}$ are given in Tables X - XXIV.

Table X. Confidence procedure results for $K = 0.1$ and $\theta_0 = 0.1$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.0920	4.5981e-3	0.0665	0.0963	1.2270	11.1111	20.9877
2	0.1155	5.7750e-3	0.0580	0.0960	1.4550	9.4094	17.9334
3	0.1042	5.2125e-3	0.0617	0.0962	1.3486	10.0100	19.0180
4	0.1155	5.7750e-3	0.0580	0.0960	1.4550	9.4094	17.9334
5	0.1261	6.3052e-3	0.0549	0.0959	1.5513	9.0090	17.2064
6	0.1363	6.8130e-3	0.0524	0.0959	1.6399	8.7087	16.6590
7	0.1461	7.3040e-3	0.0501	0.0958	1.7222	8.4084	16.1098
8	0.1556	7.7819e-3	0.0481	0.0957	1.8002	8.3083	15.9263
9	0.1650	8.2492e-3	0.0463	0.0956	1.8732	8.1081	15.5588
10	0.1741	8.7075e-3	0.0447	0.0956	1.9428	8.0080	15.3747
11	0.1832	9.1581e-3	0.0432	0.0955	2.0096	8.0080	15.3747
12	0.1920	9.6020e-3	0.0419	0.0954	2.0726	7.9079	15.1905
13	0.2008	1.0040e-2	0.0406	0.0953	2.1335	7.9079	15.1905
14	0.2094	1.0472e-2	0.0395	0.0953	2.1908	7.8078	15.0060
15	0.2180	1.0900e-2	0.0384	0.0952	2.2465	7.8078	15.0060
16	0.2265	1.1323e-2	0.0374	0.0951	2.2999	7.8078	15.0060
17	0.2348	1.1741e-2	0.0364	0.0950	2.3511	7.8078	15.0060
18	0.2431	1.2156e-2	0.0355	0.0950	2.4002	7.8078	15.0060
19	0.2513	1.2566e-2	0.0347	0.0949	2.4473	7.8078	15.0060
20	0.2595	1.2973e-2	0.0339	0.0948	2.4925	7.8078	15.0060

Table XI. Confidence procedure results for $K = 0.1$ and $\theta_0 = 0.5$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.0920	4.5981e-3	0.3000	0.4830	0.2682	8.5085	16.2931
2	0.1155	5.7750e-3	0.2621	0.4819	0.3180	7.6076	14.6365
3	0.1042	5.2125e-3	0.2787	0.4825	0.2948	7.9079	15.1905
4	0.1155	5.7750e-3	0.2621	0.4819	0.3180	7.6076	14.6365
5	0.1261	6.3052e-3	0.2485	0.4816	0.3389	7.3073	14.0806
6	0.1363	6.8130e-3	0.2369	0.4810	0.3583	7.2072	13.8950
7	0.1461	7.3040e-3	0.2268	0.4809	0.3763	7.0070	13.5230
8	0.1556	7.7819e-3	0.2179	0.4802	0.3933	7.0070	13.5230
9	0.1650	8.2492e-3	0.2100	0.4800	0.4092	6.9069	13.3368
10	0.1741	8.7075e-3	0.2027	0.4798	0.4243	6.8068	13.1503
11	0.1832	9.1581e-3	0.1962	0.4793	0.4388	6.8068	13.1503
12	0.1920	9.6020e-3	0.1901	0.4789	0.4527	6.8068	13.1503
13	0.2008	1.0040e-2	0.1846	0.4785	0.4659	6.8068	13.1503
14	0.2094	1.0472e-2	0.1794	0.4782	0.4785	6.8068	13.1503
15	0.2180	1.0900e-2	0.1746	0.4778	0.4906	6.8068	13.1503
16	0.2265	1.1323e-2	0.1701	0.4775	0.5021	6.8068	13.1503
17	0.2348	1.1741e-2	0.1658	0.4773	0.5132	6.8068	13.1503
18	0.2431	1.2156e-2	0.1618	0.4770	0.5238	6.8068	13.1503
19	0.2513	1.2566e-2	0.1581	0.4767	0.5340	6.8068	13.1503
20	0.2595	1.2973e-2	0.1545	0.4765	0.5438	6.8068	13.1503

Table XII. Confidence procedure results for $K = 0.1$ and $\theta_0 = 2$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.0920	4.5981e-3	1.1701	1.9327	0.0683	8.1081	15.5588
2	0.1155	5.7750e-3	1.0221	1.9295	0.0810	7.2072	13.8950
3	0.1042	5.2125e-3	1.0868	1.9314	0.0751	7.5075	14.4514
4	0.1155	5.7750e-3	1.0221	1.9295	0.0810	7.2072	13.8950
5	0.1261	6.3052e-3	0.9690	1.9278	0.0863	7.0070	13.5230
6	0.1363	6.8130e-3	0.9238	1.9257	0.0913	6.9069	13.3368
7	0.1461	7.3040e-3	0.8846	1.9241	0.0958	6.8068	13.1503
8	0.1556	7.7819e-3	0.8498	1.9229	0.1001	6.7067	12.9636
9	0.1650	8.2492e-3	0.8188	1.9220	0.1042	6.6066	12.7767
10	0.1741	8.7075e-3	0.7906	1.9201	0.1081	6.6066	12.7767
11	0.1832	9.1581e-3	0.7650	1.9184	0.1118	6.6066	12.7767
12	0.1920	9.6020e-3	0.7415	1.9169	0.1153	6.6066	12.7767
13	0.2008	1.0040e-2	0.7197	1.9154	0.1186	6.6066	12.7767
14	0.2094	1.0472e-2	0.6996	1.9141	0.1218	6.6066	12.7767
15	0.2180	1.0900e-2	0.6808	1.9128	0.1249	6.6066	12.7767
16	0.2265	1.1323e-2	0.6632	1.9117	0.1279	6.6066	12.7767
17	0.2348	1.1741e-2	0.6467	1.9106	0.1307	6.6066	12.7767
18	0.2431	1.2156e-2	0.6312	1.9096	0.1334	6.6066	12.7767
19	0.2513	1.2566e-2	0.6166	1.9086	0.1360	6.6066	12.7767
20	0.2595	1.2973e-2	0.6027	1.9077	0.1384	6.6066	12.7767

Table XIII. Confidence procedure results for $K = 0.1$ and $\theta_0 = 5$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.0920	4.5981e-3	2.9096	4.8326	0.0274	8.0080	15.3747
2	0.1155	5.7750e-3	2.5415	4.8228	0.0325	7.2072	13.8950
3	0.1042	5.2125e-3	2.7024	4.8275	0.0301	7.5075	14.4514
4	0.1155	5.7750e-3	2.5415	4.8228	0.0325	7.2072	13.8950
5	0.1261	6.3052e-3	2.4094	4.8185	0.0347	7.0070	13.5230
6	0.1363	6.8130e-3	2.2970	4.8160	0.0366	6.8068	13.1503
7	0.1461	7.3040e-3	2.1993	4.8122	0.0385	6.7067	12.9636
8	0.1556	7.7819e-3	2.1130	4.8093	0.0402	6.6066	12.7767
9	0.1650	8.2492e-3	2.0357	4.8042	0.0418	6.6066	12.7767
10	0.1741	8.7075e-3	1.9658	4.7995	0.0434	6.6066	12.7767
11	0.1832	9.1581e-3	1.9020	4.7984	0.0449	6.5065	12.5897
12	0.1920	9.6020e-3	1.8435	4.7946	0.0463	6.5065	12.5897
13	0.2008	1.0040e-2	1.7894	4.7911	0.0476	6.5065	12.5897
14	0.2094	1.0472e-2	1.7393	4.7878	0.0489	6.5065	12.5897
15	0.2180	1.0900e-2	1.6926	4.7848	0.0501	6.5065	12.5897
16	0.2265	1.1323e-2	1.6489	4.7820	0.0513	6.5065	12.5897
17	0.2348	1.1741e-2	1.6079	4.7793	0.0524	6.5065	12.5897
18	0.2431	1.2156e-2	1.5693	4.7734	0.0536	6.6066	12.7767
19	0.2513	1.2566e-2	1.5329	4.7709	0.0546	6.6066	12.7767
20	0.2595	1.2973e-2	1.4985	4.7687	0.0556	6.6066	12.7767

Table XIV. Confidence procedure results for $K = 0.1$ and $\theta_0 = 10$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.0920	4.5981e-3	5.8086	9.6685	0.0137	7.9079	15.1905
2	0.1155	5.7750e-3	5.0737	9.6499	0.0163	7.1071	13.7091
3	0.1042	5.2125e-3	5.3950	9.6589	0.0151	7.4074	14.2661
4	0.1155	5.7750e-3	5.0737	9.6499	0.0163	7.1071	13.7091
5	0.1261	6.3052e-3	4.8098	9.6415	0.0174	6.9069	13.3368
6	0.1363	6.8130e-3	4.5856	9.6314	0.0183	6.8068	13.1503
7	0.1461	7.3040e-3	4.3905	9.6238	0.0193	6.7067	12.9636
8	0.1556	7.7819e-3	4.2181	9.6180	0.0201	6.6066	12.7767
9	0.1650	8.2492e-3	4.0638	9.6078	0.0209	6.6066	12.7767
10	0.1741	8.7075e-3	3.9242	9.6047	0.0217	6.5065	12.5897
11	0.1832	9.1581e-3	3.7969	9.5964	0.0225	6.5065	12.5897
12	0.1920	9.6020e-3	3.6801	9.5888	0.0232	6.5065	12.5897
13	0.2008	1.0040e-2	3.5722	9.5818	0.0238	6.5065	12.5897
14	0.2094	1.0472e-2	3.4721	9.5753	0.0245	6.5065	12.5897
15	0.2180	1.0900e-2	3.3788	9.5692	0.0251	6.5065	12.5897
16	0.2265	1.1323e-2	3.2916	9.5635	0.0257	6.5065	12.5897
17	0.2348	1.1741e-2	3.2098	9.5582	0.0263	6.5065	12.5897
18	0.2431	1.2156e-2	3.1328	9.5532	0.0268	6.5065	12.5897
19	0.2513	1.2566e-2	3.0601	9.5415	0.0273	6.6066	12.7767
20	0.2595	1.2973e-2	2.9914	9.5370	0.0278	6.6066	12.7767

Table XV. Confidence procedure results for $K = 1$ and $\theta_0 = 0.1$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.2595	1.2973e-2	0.0428	0.0914	1.4990	15.1151	27.9456
2	0.4087	2.0437e-2	0.0284	0.0892	1.8922	15.0150	27.7755
3	0.3371	1.6857e-2	0.0341	0.0903	1.7328	14.7147	27.2642
4	0.4087	2.0437e-2	0.0284	0.0892	1.8922	15.0150	27.7755
5	0.4745	2.3727e-2	0.0243	0.0882	1.9941	15.6156	28.7928
6	0.5347	2.6734e-2	0.0213	0.0870	2.0534	16.5165	30.3051
7	0.5893	2.9467e-2	0.0189	0.0859	2.0752	17.4174	31.8012
8	0.6387	3.1935e-2	0.0171	0.0846	2.0722	18.5185	33.6077
9	0.6831	3.4154e-2	0.0155	0.0833	2.0494	19.7197	35.5508
10	0.7228	3.6138e-2	0.0142	0.0820	2.0094	20.9209	37.4650
11	0.7581	3.7905e-2	0.0131	0.0807	1.9597	22.2222	39.5062
12	0.7894	3.9472e-2	0.0122	0.0793	1.9007	23.5235	41.5135
13	0.8171	4.0857e-2	0.0113	0.0779	1.8388	24.9249	43.6373
14	0.8415	4.2076e-2	0.0106	0.0764	1.7769	26.4264	45.8693
15	0.8629	4.3147e-2	0.0100	0.0749	1.7129	27.9279	48.0562
16	0.8817	4.4084e-2	0.0094	0.0733	1.6483	29.4294	50.1979
17	0.8980	4.4902e-2	0.0089	0.0717	1.5883	31.0310	52.4328
18	0.9123	4.5615e-2	0.0084	0.0702	1.5257	32.5325	54.4814
19	0.9247	4.6234e-2	0.0080	0.0686	1.4693	34.1341	56.6169
20	0.9354	4.6770e-2	0.0077	0.0670	1.4157	35.7357	58.7010

Table XVI. Confidence procedure results for $K = 1$ and $\theta_0 = 0.5$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.2595	1.2973e-2	0.1814	0.4716	0.4553	8.9089	17.0241
2	0.4087	2.0437e-2	0.1250	0.4640	0.5669	9.6096	18.2958
3	0.3371	1.6857e-2	0.1476	0.4679	0.5226	9.1091	17.3885
4	0.4087	2.0437e-2	0.1250	0.4640	0.5669	9.6096	18.2958
5	0.4745	2.3727e-2	0.1086	0.4600	0.5932	10.2102	19.3779
6	0.5347	2.6734e-2	0.0961	0.4559	0.6057	10.9109	20.6313
7	0.5893	2.9467e-2	0.0862	0.4515	0.6075	11.7117	22.0518
8	0.6387	3.1935e-2	0.0782	0.4472	0.6006	12.5125	23.4594
9	0.6831	3.4154e-2	0.0715	0.4421	0.5887	13.5135	25.2009
10	0.7228	3.6138e-2	0.0659	0.4370	0.5719	14.5145	26.9223
11	0.7581	3.7905e-2	0.0611	0.4315	0.5524	15.6156	28.7928
12	0.7894	3.9472e-2	0.0569	0.4259	0.5304	16.7167	30.6389
13	0.8171	4.0857e-2	0.0533	0.4200	0.5078	17.9179	32.6253
14	0.8415	4.2076e-2	0.0501	0.4135	0.4853	19.2192	34.7447
15	0.8629	4.3147e-2	0.0472	0.4071	0.4624	20.5205	36.8301
16	0.8817	4.4084e-2	0.0446	0.4002	0.4406	21.9219	39.0381
17	0.8980	4.4902e-2	0.0423	0.3933	0.4192	23.3233	41.2069
18	0.9123	4.5615e-2	0.0403	0.3859	0.3994	24.8248	43.4869
19	0.9247	4.6234e-2	0.0384	0.3785	0.3803	26.3263	45.7219
20	0.9354	4.6770e-2	0.0366	0.3711	0.3620	27.8278	47.9118

Table XVII. Confidence procedure results for $K = 1$ and $\theta_0 = 2$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.2595	1.2973e-2	0.6441	1.9023	0.1314	7.2072	13.8950
2	0.4087	2.0437e-2	0.4514	1.8760	0.1629	8.0080	15.3747
3	0.3371	1.6857e-2	0.5288	1.8896	0.1505	7.5075	14.4514
4	0.4087	2.0437e-2	0.4514	1.8760	0.1629	8.0080	15.3747
5	0.4745	2.3727e-2	0.3949	1.8618	0.1702	8.6086	16.4761
6	0.5347	2.6734e-2	0.3515	1.8465	0.1735	9.3093	17.7520
7	0.5893	2.9467e-2	0.3170	1.8315	0.1736	10.0100	19.0180
8	0.6387	3.1935e-2	0.2887	1.8150	0.1714	10.8108	20.4529
9	0.6831	3.4154e-2	0.2652	1.7968	0.1676	11.7117	22.0518
10	0.7228	3.6138e-2	0.2452	1.7787	0.1623	12.6126	23.6344
11	0.7581	3.7905e-2	0.2281	1.7588	0.1563	13.6136	25.3739
12	0.7894	3.9472e-2	0.2131	1.7371	0.1499	14.7147	27.2642
13	0.8171	4.0857e-2	0.2000	1.7153	0.1431	15.8158	29.1302
14	0.8415	4.2076e-2	0.1885	1.6917	0.1363	17.0170	31.1382
15	0.8629	4.3147e-2	0.1781	1.6681	0.1295	18.2182	33.1174
16	0.8817	4.4084e-2	0.1689	1.6426	0.1229	19.5195	35.2289
17	0.8980	4.4902e-2	0.1605	1.6170	0.1165	20.8208	37.3066
18	0.9123	4.5615e-2	0.1529	1.5895	0.1105	22.2222	39.5062
19	0.9247	4.6234e-2	0.1460	1.5602	0.1051	23.7237	41.8193
20	0.9354	4.6770e-2	0.1397	1.5326	0.0996	25.1251	43.9375

Table XVIII. Confidence procedure results for $K = 1$ and $\theta_0 = 5$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.2595	1.2973e-2	1.5454	4.7649	0.0544	6.8068	13.1503
2	0.4087	2.0437e-2	1.0859	4.6983	0.0675	7.7077	14.8213
3	0.3371	1.6857e-2	1.2705	4.7312	0.0623	7.2072	13.8950
4	0.4087	2.0437e-2	1.0859	4.6983	0.0675	7.7077	14.8213
5	0.4745	2.3727e-2	0.9512	4.6636	0.0704	8.3083	15.9263
6	0.5347	2.6734e-2	0.8476	4.6301	0.0717	8.9089	17.0241
7	0.5893	2.9467e-2	0.7652	4.5931	0.0717	9.6096	18.2958
8	0.6387	3.1935e-2	0.6978	4.5521	0.0708	10.4104	19.7371
9	0.6831	3.4154e-2	0.6415	4.5070	0.0692	11.3113	21.3432
10	0.7228	3.6138e-2	0.5938	4.4619	0.0670	12.2122	22.9330
11	0.7581	3.7905e-2	0.5527	4.4168	0.0644	13.1131	24.5067
12	0.7894	3.9472e-2	0.5170	4.3628	0.0618	14.2142	26.4080
13	0.8171	4.0857e-2	0.4856	4.3086	0.0590	15.3153	28.2850
14	0.8415	4.2076e-2	0.4578	4.2543	0.0560	16.4164	30.1378
15	0.8629	4.3147e-2	0.4331	4.1908	0.0533	17.7177	32.2963
16	0.8817	4.4084e-2	0.4108	4.1318	0.0505	18.9189	34.2586
17	0.8980	4.4902e-2	0.3907	4.0680	0.0478	20.2202	36.3519
18	0.9123	4.5615e-2	0.3725	3.9995	0.0454	21.6216	38.5683
19	0.9247	4.6234e-2	0.3559	3.9308	0.0430	23.0230	40.7455
20	0.9354	4.6770e-2	0.3406	3.8573	0.0409	24.5245	43.0345

Table XIX. Confidence procedure results for $K = 1$ and $\theta_0 = 10$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.2595	1.2973e-2	3.0405	9.5332	0.0275	6.7067	12.9636
2	0.4087	2.0437e-2	2.1374	9.4018	0.0341	7.6076	14.6365
3	0.3371	1.6857e-2	2.5003	9.4670	0.0315	7.1071	13.7091
4	0.4087	2.0437e-2	2.1374	9.4018	0.0341	7.6076	14.6365
5	0.4745	2.3727e-2	1.8727	9.3410	0.0356	8.1081	15.5588
6	0.5347	2.6734e-2	1.6693	9.2662	0.0363	8.8088	16.8417
7	0.5893	2.9467e-2	1.5073	9.1924	0.0363	9.5095	18.1147
8	0.6387	3.1935e-2	1.3749	9.1107	0.0358	10.3103	19.5576
9	0.6831	3.4154e-2	1.2643	9.0294	0.0350	11.1111	20.9877
10	0.7228	3.6138e-2	1.1705	8.9394	0.0339	12.0120	22.5811
11	0.7581	3.7905e-2	1.0898	8.8405	0.0326	13.0130	24.3326
12	0.7894	3.9472e-2	1.0196	8.7415	0.0312	14.0140	26.0641
13	0.8171	4.0857e-2	0.9579	8.6333	0.0298	15.1151	27.9456
14	0.8415	4.2076e-2	0.9033	8.5249	0.0283	16.2162	29.8028
15	0.8629	4.3147e-2	0.8546	8.3980	0.0269	17.5175	31.9664
16	0.8817	4.4084e-2	0.8109	8.2799	0.0255	18.7187	33.9335
17	0.8980	4.4902e-2	0.7714	8.1524	0.0241	20.0200	36.0320
18	0.9123	4.5615e-2	0.7355	8.0154	0.0229	21.4214	38.2541
19	0.9247	4.6234e-2	0.7028	7.8781	0.0217	22.8228	40.4368
20	0.9354	4.6770e-2	0.6729	7.7406	0.0206	24.2242	42.5803

Table XX. Confidence procedure results for $K = 10$ and $\theta_0 = 0.1$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.9354	4.6770e-2	0.0078	0.0497	0.9273	54.5546	79.3471
2	0.9977	4.9883e-2	0.0039	0.0304	0.7094	72.4725	92.4224
3	0.9871	4.9353e-2	0.0053	0.0385	0.7994	64.8649	87.6552
4	0.9977	4.9883e-2	0.0039	0.0304	0.7094	72.4725	92.4224
5	0.9996	4.9980e-2	0.0032	0.0245	0.6559	77.9780	95.1503
6	0.9999	4.9997e-2	0.0026	0.0202	0.6257	81.9820	96.7535
7	1.0000	5.0000e-2	0.0022	0.0170	0.5996	84.8849	97.7153
8	1.0000	5.0000e-2	0.0019	0.0146	0.5813	87.0871	98.3326
9	1.0000	5.0000e-2	0.0017	0.0127	0.5662	88.7888	98.7431
10	1.0000	5.0000e-2	0.0015	0.0113	0.5399	90.0901	99.0179
11	1.0000	5.0000e-2	0.0014	0.0101	0.5317	91.1912	99.2240
12	1.0000	5.0000e-2	0.0013	0.0091	0.5242	92.0921	99.3746
13	1.0000	5.0000e-2	0.0012	0.0082	0.5428	92.8929	99.4949
14	1.0000	5.0000e-2	0.0011	0.0075	0.5231	93.4935	99.5767
15	1.0000	5.0000e-2	0.0010	0.0070	0.4960	93.9940	99.6393
16	1.0000	5.0000e-2	0.0010	0.0064	0.5157	94.4945	99.6969
17	1.0000	5.0000e-2	0.0009	0.0060	0.5112	94.8949	99.7394
18	1.0000	5.0000e-2	0.0008	0.0056	0.4646	95.1952	99.7691
19	1.0000	5.0000e-2	0.0008	0.0053	0.4487	95.4955	99.7971
20	1.0000	5.0000e-2	0.0008	0.0049	0.4650	95.7958	99.8232

Table XXI. Confidence procedure results for $K = 10$ and $\theta_0 = 0.5$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.9354	4.6770e-2	0.0386	0.3120	0.2475	40.7407	64.8834
2	0.9977	4.9883e-2	0.0196	0.1860	0.1575	65.3654	88.0044
3	0.9871	4.9353e-2	0.0261	0.2405	0.1888	54.7548	79.5287
4	0.9977	4.9883e-2	0.0196	0.1860	0.1575	65.3654	88.0044
5	0.9996	4.9980e-2	0.0157	0.1471	0.1401	72.8729	92.6412
6	0.9999	4.9997e-2	0.0130	0.1193	0.1298	78.1782	95.2381
7	1.0000	5.0000e-2	0.0111	0.0992	0.1224	81.9820	96.7535
8	1.0000	5.0000e-2	0.0097	0.0843	0.1164	84.7848	97.6850
9	1.0000	5.0000e-2	0.0086	0.0726	0.1147	86.9870	98.3066
10	1.0000	5.0000e-2	0.0077	0.0634	0.1131	88.6887	98.7205
11	1.0000	5.0000e-2	0.0070	0.0564	0.1086	89.9900	98.9980
12	1.0000	5.0000e-2	0.0064	0.0504	0.1079	91.0911	99.2063
13	1.0000	5.0000e-2	0.0059	0.0455	0.1070	91.9920	99.3587
14	1.0000	5.0000e-2	0.0055	0.0416	0.1008	92.6927	99.4660
15	1.0000	5.0000e-2	0.0051	0.0378	0.1074	93.3934	99.5635
16	1.0000	5.0000e-2	0.0048	0.0350	0.1013	93.8939	99.6272
17	1.0000	5.0000e-2	0.0045	0.0322	0.1054	94.3944	99.6858
18	1.0000	5.0000e-2	0.0042	0.0300	0.1039	94.7948	99.7291
19	1.0000	5.0000e-2	0.0040	0.0283	0.0930	95.0951	99.7594
20	1.0000	5.0000e-2	0.0038	0.0266	0.0889	95.3954	99.7880

Table XXII. Confidence procedure results for $K = 10$ and $\theta_0 = 2$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.9354	4.6770e-2	0.1508	1.4169	0.0798	31.5315	53.1207
2	0.9977	4.9883e-2	0.0776	0.8569	0.0434	59.4595	83.5646
3	0.9871	4.9353e-2	0.1028	1.1093	0.0551	46.9469	71.8537
4	0.9977	4.9883e-2	0.0776	0.8569	0.0434	59.4595	83.5646
5	0.9996	4.9980e-2	0.0622	0.6732	0.0372	68.4685	90.0576
6	0.9999	4.9997e-2	0.0517	0.5412	0.0337	74.8749	93.6873
7	1.0000	5.0000e-2	0.0443	0.4456	0.0316	79.4795	95.7891
8	1.0000	5.0000e-2	0.0386	0.3744	0.0304	82.8829	97.0700
9	1.0000	5.0000e-2	0.0343	0.3215	0.0288	85.3854	97.8641
10	1.0000	5.0000e-2	0.0308	0.2791	0.0284	87.3874	98.4092
11	1.0000	5.0000e-2	0.0279	0.2470	0.0269	88.8889	98.7654
12	1.0000	5.0000e-2	0.0255	0.2192	0.0272	90.1902	99.0377
13	1.0000	5.0000e-2	0.0235	0.1976	0.0262	91.1912	99.2240
14	1.0000	5.0000e-2	0.0218	0.1782	0.0270	92.0921	99.3746
15	1.0000	5.0000e-2	0.0203	0.1630	0.0262	92.7928	99.4806
16	1.0000	5.0000e-2	0.0190	0.1498	0.0257	93.3934	99.5635
17	1.0000	5.0000e-2	0.0178	0.1389	0.0245	93.8939	99.6272
18	1.0000	5.0000e-2	0.0168	0.1280	0.0261	94.3944	99.6858
19	1.0000	5.0000e-2	0.0159	0.1192	0.0261	94.7948	99.7291
20	1.0000	5.0000e-2	0.0151	0.1124	0.0235	95.0951	99.7594

Table XXIII. Confidence procedure results for $K = 10$ and $\theta_0 = 5$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.9354	4.6770e-2	0.3665	3.7106	0.0362	27.8278	47.9118
2	0.9977	4.9883e-2	0.1911	2.2850	0.0184	56.4565	81.0396
3	0.9871	4.9353e-2	0.2519	2.9468	0.0239	43.2432	67.7867
4	0.9977	4.9883e-2	0.1911	2.2850	0.0184	56.4565	81.0396
5	0.9996	4.9980e-2	0.1535	1.7981	0.0154	66.0661	88.4849
6	0.9999	4.9997e-2	0.1280	1.4399	0.0140	73.0731	92.7494
7	1.0000	5.0000e-2	0.1097	1.1866	0.0128	77.9780	95.1503
8	1.0000	5.0000e-2	0.0959	0.9942	0.0122	81.6817	96.6444
9	1.0000	5.0000e-2	0.0851	0.8476	0.0119	84.4845	97.5927
10	1.0000	5.0000e-2	0.0764	0.7368	0.0114	86.5866	98.2008
11	1.0000	5.0000e-2	0.0693	0.6468	0.0113	88.2883	98.6284
12	1.0000	5.0000e-2	0.0634	0.5774	0.0107	89.5896	98.9162
13	1.0000	5.0000e-2	0.0585	0.5185	0.0104	90.6907	99.1334
14	1.0000	5.0000e-2	0.0542	0.4700	0.0100	91.5916	99.2930
15	1.0000	5.0000e-2	0.0505	0.4270	0.0102	92.3924	99.4212
16	1.0000	5.0000e-2	0.0472	0.3893	0.0106	93.0931	99.5229
17	1.0000	5.0000e-2	0.0444	0.3619	0.0097	93.5936	99.5896
18	1.0000	5.0000e-2	0.0419	0.3347	0.0098	94.0941	99.6512
19	1.0000	5.0000e-2	0.0396	0.3127	0.0092	94.4945	99.6969
20	1.0000	5.0000e-2	0.0376	0.2909	0.0096	94.8949	99.7394

Table XXIV. Confidence procedure results for $K = 10$ and $\theta_0 = 10$

n	$\beta(\theta_0)$	$(5\%)\beta(\theta_0)$	θ_t	θ_c	$Slope(\theta_c)$	%C	%WC
1	0.9354	4.6770e-2	0.7151	7.5742	0.0192	26.1261	45.4265
2	0.9977	4.9883e-2	0.3768	4.7212	0.0094	54.8549	79.6192
3	0.9871	4.9353e-2	0.4948	6.0514	0.0125	41.5415	65.8261
4	0.9977	4.9883e-2	0.3768	4.7212	0.0094	54.8549	79.6192
5	0.9996	4.9980e-2	0.3035	3.7201	0.0079	64.7648	87.5848
6	0.9999	4.9997e-2	0.2536	2.9853	0.0071	71.9720	92.1443
7	1.0000	5.0000e-2	0.2175	2.4502	0.0066	77.1772	94.7912
8	1.0000	5.0000e-2	0.1903	2.0560	0.0062	80.9810	96.3828
9	1.0000	5.0000e-2	0.1690	1.7534	0.0060	83.8839	97.4027
10	1.0000	5.0000e-2	0.1519	1.5222	0.0058	86.0861	98.0640
11	1.0000	5.0000e-2	0.1379	1.3423	0.0055	87.7878	98.5086
12	1.0000	5.0000e-2	0.1263	1.1838	0.0056	89.2893	98.8528
13	1.0000	5.0000e-2	0.1164	1.0661	0.0053	90.3904	99.0766
14	1.0000	5.0000e-2	0.1079	0.9595	0.0055	91.3914	99.2589
15	1.0000	5.0000e-2	0.1005	0.8735	0.0054	92.1922	99.3904
16	1.0000	5.0000e-2	0.0941	0.8081	0.0048	92.7928	99.4806
17	1.0000	5.0000e-2	0.0885	0.7433	0.0048	93.3934	99.5635
18	1.0000	5.0000e-2	0.0834	0.6889	0.0047	93.8939	99.6272
19	1.0000	5.0000e-2	0.0789	0.6351	0.0052	94.3944	99.6858
20	1.0000	5.0000e-2	0.0749	0.5915	0.0053	94.7948	99.7291

APPENDIX D

MATLAB PROGRAM - SLOPE

```
% Jeroen Stedehouder
% May 2009
% MS Texas A&M University
% Assume alpha = 0.05, where erfinv(0.5-alpha) = 1.645
clear all;
orient portrait;
set(0,'defaultaxesfontsize',8);
k = 0.1;        % SNR
theta_0 = 0.1;  % Nominal
sum = 0;
theta = 0.000000000000000000000000000000000000000000000000000001;

for n=1:20
    for i=1:n
        sum = sum + (cos((2*pi*i)/n))^2;
    end

    for j=1:20001
        slope = sqrt((((theta_0)*(1.645)^2)/(2*pi))*...
            (exp(-(((sqrt(theta_0/theta))*(1.645))-...
            (sqrt(k)*sqrt(sum)))^2)/((4*theta+k)*(theta^2))));
        X(j)= theta;
        Y(j)= slope;
        theta = theta + 0.001;
    end

    subplot(5,4,n);
    plot(X,Y);
    axis square;
    xlabel('\theta');
    ylabel('Slope(\theta)');
    title(['n = ',int2str(n)]);
    sum = 0;
    theta = 0.000000000000000000000000000000000000000000000000000001;
end
```

APPENDIX E

MATLAB PROGRAM - DECISION SURFACE

```
% Jeroen Stedehouder
% May 2009
% MS Texas A&M University
% Assume alpha = 0.05, where erfinv(0.5-alpha) = 1.645
clear all;
orient portrait;
set(gcf, 'PaperPositionMode', 'auto');
k = 1;       % SNR
theta_0 = 5; % Nominal
theta = 5;   % Theta equal to nominal
sum = 0;
x = -60;
y = -50;

for n=1:1
    for i=1:n
        sum = sum + (cos((2*pi*i)/n))^2;
    end

    for j=1:25000
        A(j) = (1/(sqrt(2*pi*k*(theta_0)^2*sum)))*exp(-(x^2)/...
            (2*k*(theta_0)^2*sum));
        B(j) = (1/(sqrt(2*pi*k*(theta_0)*theta*sum)))*...
            exp(-((y-k*sqrt((theta_0)*theta)*sum)^2)/...
            (2*k*(theta_0)*theta*sum));
        U(j) = x;
        V(j) = y;
        x = x + 0.01;
        y = y + 0.01;
    end
end

plot(U,A,'r',V,B,'g');
axis([-25 25 0 0.1]);
xlabel('x');
ylabel('p(N, S+N)');
```

APPENDIX F

MATLAB PROGRAM - SLOPE ANALYSIS FOR LARGE VARIANCE

```
% Jeroen Stedehouder
% May 2009
% MS Texas A&M University
% Assume alpha = 0.05, where erfinv(0.5-alpha) = 1.645
clear all;
format long g;
k = 0.1;      % SNR
sum = 0;
theta = 0.000000000000000000000000000000000000000000000000000001;
fid = fopen('SlopeTail.txt', 'w');
fprintf(fid, 'k = %3.1g, theta: 10-15\n',k);
theta_0 = 0.1; % Nominal
fprintf(fid, 'theta_0 = %3.1g\n',theta_0);

for n=1:20
    for i=1:n
        sum = sum + (cos((2*pi*i)/n))^2;
    end

    for j=1:15001
        lnslope = log(sqrt((((theta_0)*(1.645)^2)/(2*pi))*...
            (exp(-((((sqrt(theta_0/theta))*(1.645))-...
            (sqrt(k)*sqrt(sum)))^2)/((4*theta+k)*(theta^2)))));
        X(j)= theta;
        Y(j)= lnslope;
        theta = theta + 0.001;
    end
    delta_theta = 5;
    delta_lnslope = Y(15001)-Y(10001);
    dtheta_dlnslope = delta_lnslope/delta_theta;
    fprintf(fid, '%9.7f\n', dtheta_dlnslope);
    sum = 0;
    theta = 0.000000000000000000000000000000000000000000000000000001;
end

fclose(fid);
```

APPENDIX G

MATLAB PROGRAM - SLOPE LIKELIHOOD

```
% Jeroen Stedehouder
% May 2009
% MS Texas A&M University
% Assume alpha = 0.05, where erfinv(0.5-alpha) = 1.645
clear all;
format long g;
k = 0.1;       % SNR
theta_0 = 0.1; % Nominal
theta = 0.1;   % Theta equal to nominal
sum = 0;
sol_theta = 0;
c=0;
e=0;
f=theta_0-0.01;
count = 0;
weighted_points = 0;

for n=1:20
    for i=1:n
        sum = sum + (cos((2*pi*i)/n))^2;
    end
    % Calculating slope at theta = theta_0
    slope = sqrt((((theta_0)*(1.645)^2)/(2*pi))*...
        (exp(-((((sqrt(theta_0/theta))*(1.645))-...
        (sqrt(k)*sqrt(sum)))^2)/((4*theta+k)*(theta^2)))));
    slope_ans(n) = slope;

    % Using bisection method in order to find the theta with the same
    % slope as theta_0
    [c]=bisect(e,f,0.00000001,theta_0,k,sum,slope);
    theta_t(n) = c;

    % Calculating the 1000 thetas and their corresponding
    % slope values
    theta_range = theta_t(n);
    theta_diff = (theta_0 - theta_t(n))/999;
    for p=1:1000
        slope_repeat(p)= sqrt((((theta_0)*(1.645)^2)/(2*pi))*...
            (exp(-((((sqrt(theta_0/theta_range))*(1.645))-...
            (sqrt(k)*sqrt(sum)))^2)/((4*theta_range+k)*...
            (theta_range^2)))));
        theta_range = theta_range + theta_diff;
    end
```

```
% Calculating the weight factor for each theta interval between
% theta_range and theta_range + theta_diff
theta_range = theta_t(n);     % Resetting theta_range
for m=1:1000
    weight_factor(m) = (1/16*((4*(theta_range + theta_diff)...
        +k)/(theta_range + theta_diff))^(1/2)*(theta_range + ...
        theta_diff)*(8*(4*(theta_range + theta_diff)^2+...
        (theta_range + theta_diff)*k)^(1/2)+...
        k*log(1/8*4^(1/2)*k+4^(1/2)*(theta_range + ...
        theta_diff)+(4*(theta_range + theta_diff)^2+...
        (theta_range + theta_diff)*k)^(1/2))*...
        4^(1/2))/((theta_range + theta_diff)*(4*(theta_range +...
        theta_diff)+k))^(1/2))- (1/16*((4*theta_range+k)/...
        theta_range)^(1/2)*theta_range*(8*(4*theta_range^2+...
        theta_range*k)^(1/2)+k*log(1/8*4^(1/2)*k+4^(1/2)*...
        theta_range+(4*theta_range^2+theta_range*k)^(1/2))*...
        4^(1/2))/(theta_range*(4*theta_range+k))^(1/2));
    theta_range = theta_range + theta_diff;
end

% Calculating the maximum slope value for n
slope_max(n) = max(slope_repeat);

% Creating bins
slope_diff(1) = slope_ans(n);
for s=2:101
    slope_diff(s) = slope_ans(n) +...
        ((slope_max(n) - slope_ans(n))*((s-1)/100));
end

% Search procedure for finding which previously calulated slope
% points fall into a bin interval, then putting the slope's
% corresponding weight into this bin
for j=1:100
    for p=1:1000
        if ((slope_repeat(p) >= slope_diff(j)) &&...
                (slope_repeat(p) < (slope_diff(j+1))) && ...
                (j <= 99));
            count = count + 1;
            weighted_points = weighted_points + weight_factor(p);

        elseif ((slope_repeat(p) >= slope_diff(j)) &&...
                (slope_repeat(p)<= (slope_diff(j+1))) &&...
                (j == 100));
            count = count + 1;
            weighted_points = weighted_points + weight_factor(p);
        end
    end
    X(j)=slope_diff(j+1);
    num_points(j) = count;
```

```
        w_sum(j) = weighted_points;
        count = 0;
        weighted_points = 0;
    end

    % Plot results
    subplot(5,4,n);
    bar1=bar(X,w_sum,'histc');
    set(bar1,'FaceColor','b','EdgeColor','b')
    axis square;
    axis tight;
    v = axis;
    %l1=floor(v(1)*100);
    %l1=l1/100;
    %l2=ceil(v(2)*100);
    %l2=l2/100;
    %axis([l1 l2 0 v(4)]);
    xlabel('Slope(\theta)');
    ylabel('Likelihood');
    title(['n = ',int2str(n)]);
    sum = 0;
    slope = 0;
end


% Function for bisection method
function [c]=bisect(a,b,delta,theta_0_1,k_1,sum_1,slope_1)
ya=sqrt((((theta_0_1)*(1.645)^2)/(2*pi))*...
    (exp(-(((sqrt(theta_0_1/a))*(1.645))-(sqrt(k_1)*...
    sqrt(sum_1)))^2)/((4*a+k_1)*(a^2)))) - slope_1;
yb=sqrt((((theta_0_1)*(1.645)^2)/(2*pi))*...
    (exp(-(((sqrt(theta_0_1/b))*(1.645))-(sqrt(k_1)*...
    sqrt(sum_1)))^2)/((4*b+k_1)*(b^2)))) - slope_1;

if ya*yb > 0
    return
end
max1=1+round((log(b-a)-log(delta))/log(2));
for k=1:max1
    c=(a+b)/2; % a and b are the left and right endpoints
               % respectively
    yc=sqrt((((theta_0_1)*(1.645)^2)/(2*pi))*...
        (exp(-(((sqrt(theta_0_1/c))*(1.645))-(sqrt(k_1)*...
        sqrt(sum_1)))^2)/((4*c+k_1)*(c^2)))) -slope_1;
    if yc==0
        a=c;
        b=c;
    elseif yb*yc>0
        b=c;
        yb=yc;
    else
```

```
        a=c;
        ya=yc;
    end
    if b-a < delta % Tolerance
        return
    end
end
c=(a+b)/2;
%yc=sqrt((((theta_0_1)*(1.645)^2)/(2*pi))*...
%    (exp(-(((sqrt(theta_0_1/c))*(1.645))-(sqrt(k_1)*...
%    sqrt(sum_1)))^2)/((4*c+k_1)*(c^2)))) - slope_1;
%err=abs(b-a); % Error estimate
```

APPENDIX H

MATLAB PROGRAM - SLOPE LIKELIHOOD RATIO

```
% Jeroen Stedehouder
% May 2009
% MS Texas A&M University
% Assume alpha = 0.05, where erfinv(0.5-alpha) = 1.645
clear all
format long g
k = 0.1;       % SNR
theta_0 = 0.1; % Nominal
theta = 0.1;   % Theta equal to nominal
sum = 0;
c=0;
e=0;
f=theta_0-0.01;
A_1=0;
A_3=0;
count = 0;
weighted_points = 0;
fid = fopen('SLRatio.txt', 'w');
fprintf(fid, 'SLRatio:\n');

for n=1:20
    for i=1:n
        sum = sum + (cos((2*pi*i)/n))^2;
    end
    % Calculating slope at theta = theta_0
    slope = sqrt((((theta_0)*(1.645)^2)/(2*pi))*...
        (exp(-(((sqrt(theta_0/theta))*(1.645))-(sqrt(k)*...
        sqrt(sum)))^2)/((4*theta+k)*(theta^2))));
    slope_ans(n) = slope;

    % Using bisection method in order to find the theta with the same
    % slope as theta_0
    [c]=bisect(e,f,0.00000001,theta_0,k,sum,slope);
    theta_t(n) = c;

    % Calculating the 1000 thetas and their corresponding
    % slope values
    theta_range = theta_t(n);
    theta_diff = (theta_0 - theta_t(n))/999;
    for p=1:1000
        slope_repeat(p)= sqrt((((theta_0)*(1.645)^2)/(2*pi))*...
            (exp(-(((sqrt(theta_0/theta_range))*(1.645))-...
            (sqrt(k)*sqrt(sum)))^2)/((4*theta_range+k)*...
```

```
        (theta_range^2))));
    theta_range = theta_range + theta_diff;
end

% Calculating the weight factor for each theta interval between
% theta_range and theta_range + theta_diff
theta_range = theta_t(n);     % Resetting theta_range
for m=1:1000
    weight_factor(m) = (1/16*((4*(theta_range + theta_diff)...
        +k)/(theta_range + theta_diff))^(1/2)*(theta_range + ...
        theta_diff)*(8*(4*(theta_range + theta_diff)^2+...
        (theta_range + theta_diff)*k)^(1/2)+...
        k*log(1/8*4^(1/2)*k+4^(1/2)*(theta_range + ...
        theta_diff)+(4*(theta_range + theta_diff)^2+...
        (theta_range + theta_diff)*k)^(1/2))*...
        4^(1/2))/((theta_range + theta_diff)*(4*(theta_range +...
        theta_diff)+k))^(1/2))- (1/16*((4*theta_range+k)/...
        theta_range)^(1/2)*theta_range*(8*(4*theta_range^2+...
        theta_range*k)^(1/2)+k*log(1/8*4^(1/2)*k+4^(1/2)*...
        theta_range+(4*theta_range^2+theta_range*k)^(1/2))*...
        4^(1/2))/(theta_range*(4*theta_range+k))^(1/2));
    theta_range = theta_range + theta_diff;
end

% Calculating the maximum slope value for n
slope_max(n) = max(slope_repeat);

% Creating bins
slope_diff(1) = slope_ans(n);
for s=2:101
    slope_diff(s) = slope_ans(n) + ...
        ((slope_max(n) - slope_ans(n))*((s-1)/100));
end

% Search procedure for finding which previously calulated slope
% points fall into a bin interval, then putting the slope's
% corresponding weight into this bin
for j=1:100
    for p=1:1000
        if ((slope_repeat(p) >= slope_diff(j)) &&...
                (slope_repeat(p) < (slope_diff(j+1))) && ...
                (j <= 99));
            count = count + 1;
            weighted_points = weighted_points + weight_factor(p);

        elseif ((slope_repeat(p) >= slope_diff(j)) &&...
                (slope_repeat(p) <= (slope_diff(j+1))) &&...
                (j == 100));
            count = count + 1;
            weighted_points = weighted_points + weight_factor(p);
        end
```

```
        end

        X(j)=j;
        Y(j)=slope_diff(j+1);
        num_points(j) = count;
        w_sum(j) = weighted_points;
        count = 0;
        weighted_points = 0;
    end

    cutoff_min = 25;
    cutoff_max = 75;

    % Calculating area A_1
    for w=1:cutoff_min
        width(w) = slope_diff(w+1)-slope_diff(w);
        A_1 = A_1 + (width(w)*w_sum(w));
    end

    % Calculating area A_3
    for w=(cutoff_max+1):100
        width(w) = slope_diff(w+1)-slope_diff(w);
        A_3 = A_3 + (width(w)*w_sum(w));
    end

    A_ratio=A_1/A_3;
    fprintf(fid, '%g %6.3f\n',n, A_ratio);
    sum = 0;
    slope = 0;
    A_1=0;
    A_3=0;
    A_ratio=0;
end
fclose(fid);
```

APPENDIX I

MATLAB PROGRAM - CONFIDENCE LEVEL

```
% Jeroen Stedehouder
% May 2009
% MS Texas A&M University
% Assume alpha = 0.05, where erfinv(0.5-alpha) = 1.645
clear all;
format long g;
k = 0.1;       % SNR
theta_0 = 0.1; % Nominal
theta = 0.1;   % Theta equal to nominal
sum = 0;
a=0;
c=0;
e=0;
f=theta_0-0.01;
theta_c=0;
slope_c=0;
fid = fopen('Conf.txt', 'w')
fprintf(fid, 'For k = %g and theta = %g:\n\n', k, theta)

for n=1:20
    for i=1:n
        sum = sum + (cos((2*pi*i)/n))^2;
    end
    slope = sqrt((((theta_0)*(1.645)^2)/(2*pi))*...
        (exp(-(((sqrt(theta_0/theta))*(1.645))-(sqrt(k)*...
        sqrt(sum)))^2)/((4*theta+k)*(theta^2))));
    slope_min(n) = slope;

    % Using bisection method in order to find the theta with the
    % same slope as theta_0
    [c]=bisect(e,f,0.00000001,theta_0,k,sum,slope);
    theta_t(n) = c;

    % Calculating the 1000 thetas and their corresponding
    % slope values
    theta_range(1) = theta_t(n);
    theta_diff = (theta_0 - theta_t(n))/999;
    for p=1:1000
        slope_repeat(p)= sqrt((((theta_0)*(1.645)^2)/(2*pi))*...
            (exp(-(((sqrt(theta_0/theta_range(p)))*(1.645))-...
        (sqrt(k)*sqrt(sum)))^2)/((4*theta_range(p)+k)*...
        (theta_range(p)^2))));
        theta_range(p+1) = theta_range(p) + theta_diff;
```

```
end
theta_range(1001)=[];

% Calculating the maximum slope and delta value for n
slope_max(n) = max(slope_repeat);
slope_range(n) = slope_max(n) - slope_min(n);

% Calculating performance
a = (sqrt(theta_0/theta)*1.645)-(sqrt(k)*sqrt(sum));
syms x;
beta(n) = double(int((1/sqrt(2*pi))*exp((-(x^2))/2), x, a,...
    inf));

delta_beta5(n) = 0.05 * beta(n);

% Flip the vectors in order to calculate values from
% theta_0 to theta_t(n)
slope_repeat=slope_repeat(end:-1:1);
theta_range=theta_range(end:-1:1);
for p=1:999
    % Making sure that when theta_range goes beyond maximum
    % slope, then dSdT can not change more than the maximum
    % slope
    if(slope_repeat(p) <= slope_repeat(p+1))
        dSdT(p)=(theta_0-theta_range(p))*slope_repeat(p);
    else
        dSdT(p)=(theta_0-theta_range(p))*slope_max(n);
    end
end
%For p=1000, this equation is always satisfied
dSdT(1000)=(theta_0-theta_range(1000))*slope_max(n);

for p=1:999
    if ((dSdT(p) <= delta_beta5(n)) && (dSdT(p+1) >= ...
            delta_beta5(n)))
        theta_c=theta_range(p);
        slope_c=slope_repeat(p);
    end
end
conf(n)=((theta_0-theta_c)/(theta_0-theta_t(n)))*100;

% Calculating the ratio of the areas under triangle
slope_triangle(n)=(slope_min(n))/(theta_0-theta_t(n));
constant_triangle(n)=-(slope_triangle(n)*theta_t(n));
thetahat_height_triangle(n)=slope_triangle(n)*theta_c+...
    constant_triangle(n);
sq_area_triangle(n)=(theta_0-theta_c)*...
    thetahat_height_triangle(n);
tr_area_triangle(n)=0.5*((slope_min(n)-...
    thetahat_height_triangle(n))*(theta_0-theta_c));
tot_area(n)=sq_area_triangle(n)+tr_area_triangle(n);
```

```
    tot_area_triangle(n)=0.5*(slope_min(n)*(theta_0-theta_t(n)));
    weighted_conf(n)=(tot_area(n)/tot_area_triangle(n))*100;

    % Outputting the minimum, maximum and delta(min,max) to a
    % file
    fprintf(fid,...
        '%g &%6.4f &%6.4e &%6.4f &%6.4f &%6.4f &%6.4f \n',n,...
        beta(n), delta_beta5(n),theta_c,slope_c,conf(n),...
        weighted_conf(n));

    sum = 0;
    slope = 0;
    a=0;
    theta_c=0;
    slope_c=0
end
fclose(fid);
```

VITA

Jeroen Stedehouder was born in Gorinchem, Netherlands and moved to Texas at the age of 16. He attended Southwest Texas Junior College where he was inducted into Phi Theta Kappa Honor Society. He later transferred to Texas A&M University and received his Bachelor of Science in computer engineering in May 2006. He then joined the electrical engineering Signal Processing and Telecommunication group shortly after. His current research interests include signal detection and estimation. He plans to continue his education by obtaining a second master's degree in financial mathematics. Jeroen Stedehouder may be reached at Department of Electrical and Computer Engineering, Texas A&M University, 214 Zachry Engineering Center, College Station, TX 77843-3128, U.S.A. He can also be contacted via email at jeroen@tamu.edu.