A NOVEL CHARACTERIZATION OF SIGNAL SPACE DISTORTION AND ITS APPLICATION TO THE QUANTIFICATION OF LOCALIZATION ALGORITHM ERROR

By

JOHN-AUSTEN FRANCISCO

A dissertation submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Computer Science

written under the direction of

Richard P. Martin

and approved by

New Brunswick, New Jersey

October, 2015

ABSTRACT OF THE DISSERTATION

A Novel Characterization of Signal Space Distortion and its Application to the Quantification of Localization Algorithm Error

By JOHN-AUSTEN FRANCISCO

Dissertation Director:

Richard P. Martin

The constant reduction in cost and increase in the power of computing machinery has resulted in an ever-increasing interest and deployment of Internet-enabled sensing systems. Such systems have the distinctly difficult task of making use of noisy data sampled from dynamic environments. While some natural processes may have very exact theoretical models, real actual data rarely holds to the model, making the comparison, improvement and iterative development of sensing applications extremely difficult. The inability to determine whether a sensing system's error is the result of noisy data or algorithmic miscomputation, or the prevalence and significance of signal errors in a particular environment make the causes of the error inscrutable. In such cases strongly amortized or very general probabilistic analysis is often used as a last resort resulting in conclusions that are overly generic, heuristic, or strongly underdetermined. We present a systematic method that can be used to construct a holistic synthetic error model for sensed data, the algorithms that process it and the environment in which it is sampled. We demonstrate how this method can be applied to the problem of laterative localization to construct deductive, analytic and evaluative mechanisms that allow model misperception, algorithmic error and environmental character to be understood.

Acknowledgments

I would like to acknowledge my advisor, Prof. Richard P. Martin, for the many direct and indirect lessons I have learned from him over the years.

I have found Rich Howard, Chung-Chieh Shan and Shridatt Sugrim to all be very knowledgeable, good at explaining things, incredibly generous with their time, and inordinately patient. I would be considerably diminished if not for their intellectual largesse.

I would like to thank John McCabe, Thomas Moffett, Brian Sutow and Thomas Walsh for their varied help, support and advice over the years. Even though it may not have been apparent, you have all been instrumental in your own ways. I would also like to recognize Tom Scharpling for his constant inspiration and example.

Finally, I am particularly grateful to my mother for her unflagging support and understanding as well as instilling in me a particular singleminded determination and a true appreciation for learning.

Dedication

For everyone who took the hard road,

because there was nothing left to find on the well-trod path.

Table of Contents

List of Tables

List of Figures

Chapter 1
Introduction

Over the past decade the constant reduction in cost and increase in the power of computing machinery has made sensors widely available and the devices that mount them quite capable, cheap, and long-lasting. Coupled with the ubiquity of wireless networks, Internet-enabled sensing systems are more widely deployed and available than ever. Such systems have the distinctly difficult task of making use of noisy data sampled from dynamic environments, in particular, radio signal strength.

While some natural processes may have very exact theoretical models, data sampled from a live environment can diverge widely. The environment forms a noise manifold that distorts what the process' model would predict, making it necessary to model not only the process itself, but also the way the environment affects it. In some cases, environmental effects can be particularly strong and difficult to model, as is the case when modeling the radio signal strength to distance relation indoors for 802.11 data radio.

Sampled data might contain any amount of noise of multiple types. In order to bound the effects of the noise, both the noise type and its strength must be determined. The main complicating factor is aliasing between these dimensions. Being able to determine the exact type and strength of noise components is tantamount to being able to predict the configuration of the noise environment from moment to moment, which is infeasible at best, if not impossible. The inability to gauge the amount of noise in sampled radio signal strength data makes it difficult to determine whether a sensing system's error is the result of noisy data or algorithmic miscomputation. Since the environments that such systems work on are often strongly underdetermined, interpreting, comparing and reasoning about results is only done on a heuristic level.

Systems that compute on environmental data fall in to one of two general classes; pointwise and model-based. Pointwise systems record data under known conditions, match live data against it, and construe the governing conditions of the live data to be those that correspond to the recorded data that is most similar to the live data. Such systems have error residuals that deal directly with variations in sampling for each configuration of known conditions, which tends to make them readily calculable. Pointwise systems however can not generalize beyond the points at which they were sampled. We consider chiefly the second class, algorithms that are model-based.

Model-based systems are a step more abstract. Such systems rely on a model that describes the conditional relation between sampled data and some set of governing values or environmental conditions. The first stage of such a system is to collect a descriptive set of data recorded under known conditions. The model is then parameterized to produce the recorded conditions for each recorded data sample with a minimum of error. The model is then assumed to be tuned to the environment and is used to determine the conditions under which live data samples were collected. Model-based algorithms can report conditions that were not included in the direct sample set. While use of a model makes such algorithms much more flexible, it complicates the calculation of a residual since the method of parameterizing the model is a source of error as well.

Due to this extra layer of estimation the precise cause any of model-based system's result is doubly obscured, causing such results to be, at best, unparameterized metrics. Not having a clear and exact cause for each result has several deleterious effects. It is not clear how to improve model-based algorithms since noise effects are inseparable from algorithmic error. It is not clear how to compare results of algorithms computed in different environments, since the effect of the environment on an algorithm's results are unknown. It is not clear how to determine a reasonable a-priori error bound for an algorithm's performance in an environment, since the type, degree and strength of distortion and noise processes is not readily estimable.

There has been a large amount of work done, based on reasonable assumptions an extrapolations, to analyze the signal environment, identify signal distortions or benchmark algorithms, but as isolated problems. To date there has been very little work done to construct a direct causal link between characteristics of the environment and algorithmic error, which requires an end-to-end model describing not only all of these issues, but how they are linked. We define a series of mechanisms that together can be used to construct a synthetic error model to address these shortcomings of underdetermined model-based systems. We define a deductive method to determine the dominant distortion type and strength from any signal vector. We define an analytic method to determine how error types affect an algorithm. Finally we define an evaluative method to determine the type and strength of error in an environment and bound system performance. In order to illustrate the efficacy of these mechanisms, we apply them to the underdetermined model-based system of indoor laterative radio localization.

Radio localization is the process of locating a device based on how its radio signals are affected by its environment. Each physical location's radio properties are distinctive, but these properties fluctuate, making recorded signal information difficult to reason about. Being able to pair one group of radio properties to one exact location would allow a system to locate any transmitter in an indoor environment. The Received Signal Strength Indicator (RSSI) is a value determined by a wireless networking card that qualitatively describes how strong a signal is. This metric is often used for localization since it is readily accessible for minimal effort, is computed by the card itself and is related to the distance a signal has traveled.

Pointwise localization algorithms are referred to as 'scene-matching' and model-based ones as 'laterative'. Scene-matching and laterative algorithms both have an offline training phase followed by an online testing phase wherein devices are localized. During the training phase a training device transmits from known locations. These transmissions are sensed by multiple landmark (LM) devices. Landmarks are devices at other known locations that monitor the signal space and record the strength of all transmissions they detect. A strength indicator from each LM that hears a given transmission is arranged in a vector called a fingerprint. Fingerprints are the fundamental quanta of information for localization algorithms. As part of the training or testing phase the algorithm determines governing parameters from the fingerprints recorded during the training phase (training data). During the testing phase the algorithm uses the computed parameters to determine the location of the source of the fingerprint to be located (testing data) by either converting the testing data directly to a vector of distances or using it as divergence indicators analogous to distances.

The signal-to-distance model lies at the heart of laterative localization, yet it is strongly underdetermined. Consumer data radios roughly estimate received signal strength (RSS) as a byproduct of decoding a signal as they often do not need an exact measurement to meet data networking standards. Each manufacturer's device also reports a received signal strength indicator (RSSI) to the user based on its own RSS discretization function, which may or may not resemble other manufactuers' functions. Due the exigencies of its calculation, RSSI is naturally a particularly amorphous value.

Further complicating the use of RSSIs is the indoor signal environment. The indoor signal environment is very fluid, containing many objects and events that cause non-uniform signal propagation that can vary quickly in intensity and character. The range of frequencies outlined in the 802.11 standards are fairly high, causing their corresponding wavelengths to be similar to the size of many of the objects that commonly occur indoors. The size and number of such objects causes the radio energy to scatter, bounce and recombine rather than penetrate or be fully occluded. While this effect is excellent for coverage of an area, it is particularly deleterious when using a signal's strength to determine distance as it is extremely likely that the signal did not travel directly from the transmitter to the receiver in a straight line, free of obstructions, losing power only while propagating freely through space.

Since RSSI is an inexact measure and the true conditions that caused a given live RSSI are unknown, it is not clear if a laterative localization algorithm's error is due to model parameterization or algorithmic error, clearly demonstrating the first of the three shortcomings of underdetermined model-based systems. Since algorithmic error can not be isolated from parameter estimation error, algorithmic error is singular per experiment. Since error trends can not be determined, no reasoned improvements can be made to laterative localization algorithms, demonstrating the second shortcoming of underdetermined model-based systems. Since the influence of the environment on algorithmic error can not be isolated, algorithmic results between environments can not be compared and no a-priori estimates of algorithm performance can be made based on an environment's characteristics, demonstrating the third shortcoming of underdetermined model-based systems. The fundamental cause of each shortcoming is the lack of an error model for the process in question.

Below we will demonstrate how to construct a synthetic error model that can serve to provide the critical context in which to interpret and assess the results of different stages of laterative localization systems. This model will enable the generation of deductive, analytic and evaluative mechanisms to combat each shortcoming listed above, respectively.

1.1 Contributions

Without an understanding of the context in which values are computed it is not possible to determine their significance. Laterative localization systems compute in such a rational vacuum as they rest on a basis of solecistic presumption. It is the purpose of this thesis to systematically construct methods to deduce, analyze and predict radio distortion parameters, the effect of distortion on localization error and the average error of an algorithm in an environment, respectively.

In Chapter 3 we define the Discretely Distributed Log-Hölder Metric (DDLHM), which we use to characterize the manner and degree of distortion in 802.11 signal strengths relative to the lognormal path model. We then use these characterizations to detect the disposition of the dominant distortion in an exhaustive suite of synthetic signal vectors. We then determine a reduced set of maximally diagnostic distortion parameters. Using only 4% of the exhaustive set, our method detects the exact distortion profile 76% of the time, and matches with a small degree of error 85% of the time, and with an acceptable degree of error 95& of the time.

In Chapter 4 we construct and demonstrate an analytic process that we employ to benchmark algorithmic performance in the presence of distortions. We verify our findings on both point-based and laterative algorithms and identify unintuitive error trends in algorithmic response graphs. In order to qualify algorithmic robustness we set forward rotated trilateration as a standard comparative laterative localization algorithm that directly translates signal to distance estimation error to localization error.

In Chapter 5 we apply our techniques from Chapter 3 to identify distortion types in three different physical environments and determine the likely performance of several algorithms based on their benchmarks as computed in Chapter 4. For all environments and algorithms but our own diagnostic algorithm specifically engineered to have extreme error sensitivity, we can predict the average localization error to within 2 meters of the actual leave-one-out localization results, and in most cases to within 1 meter.

Chapter 2
Background and Previous Work

There are many systems developed for and continuing research into localization of a variety of objects and events using many discretizing technologies, such as angle of arrival (AoA) [29], time of arrival (ToA) [41], and connection [11, 25]. Laterative localization is a perennial favorite and has been the basis of many systems. Lateraitve localization systems have employed several signal strength-based ranging techniques, such as: ultrasound [30, 38], infrared [37], and various radio frequencies [26, 4, 28]. Localization is hardly restricted to only 802.11 radio, however it remains extremely popular as a ranging mechanic due to several distinct advantages.

Due to its nearly universal acceptance as an end user networking standard, 802.11 networks can be presumed present in nearly any indoor environment whether it is a residence, business, or industrial facility. Since it is so widely deployed, 802.11 hardware is readily available for many interface types at consumer prices. Beyond computers, 802.11 is standard in many personal electronic devices like tablets and smartphones, enabling tracking and workflow applications. Retail 802.11 networking hardware can be easily re-purposed for RSS data collection. Many 802.11 wireless drivers can be put into a 'monitor' mode that enables a device to see any traffic on a given channel, not just traffic addressed to the device. This mode makes sensing a client device's transmissions trivial and requires no collusion or cooperation on the part of the device being located, allowing very simple devices or devices that no central authority has control over to be located with no impact on an existing data network. Due to the low cost of the hardware, ease of repurposing and ubiquitous deployment creating a large target community, indoor localization is often researched with 802.11 hardware.

Despite its distinct advantages, popularity and inclusion in many laterative localization systems, 802.11 signal strength ranging presents some unique difficulties. One of its most fundamental problems is the disconnect between localization algorithm performance and the disposition of the signal space. As P. Bahl and V. N. Padmanabhan stated in their seminal 2000 paper, *RADAR: An In-Building RF-Based User Location and Tracking System*:

> For a radio channel, signal propagation in an indoor environment is dominated by reflections, diffraction, and scattering of radio waves caused by structures within the building.

> The transmitted signal generally reaches the receiver via multiple paths ... since multipath within buildings is strongly influenced by the layout of the building, the construction material used, and the number and type of objects in the building, characterizing the radio channel in such an environment is challenging. ([9])

The signal model used is pivotal to any laterative localization system. Signal strength measures are often the only indication of an unknown transmitter's location. Being able to fully understand why a given signal was sensed is tantamount to computing how the transmitter fits into the signal and physical environments. Determining this relationship is certainly difficult because any movement from a sampled point indoors to a new point is another twist of the kaleidoscope of propagation mechanics. While it is convenient to presume that all indoor propagation paths will be mostly lognormal with relatively little difference. We maintain that in reality this is largely untrue, as recent investigations have begun to bear out. [32, 10, 7] Recently M. Ficco, C. Esposito and A. Napolitano stated in their 2014 paper, *Calibrating Indoor Positioning Systems with Low Efforts*:

"Indoor signal propagation suffers from multipath fading effects, due to reflection, diffraction, and refraction caused by met obstacles. Such effects cause severe consequences on the propagation characteristics and on the model to be utilized." ([22])

Noting, with almost the exact same language, the same difficulties that the authors of the *RADAR* paper experienced. Despite 14 years of active research into localization algorithms, there has been little real progress in determining why localization algorithms perform as they do. While research is done based on reasonable assumptions and extrapolations, there is been precious little work done not just to analyze the signal environment, identify signal distortions and benchmark algorithms, but to build a direct causal link between characteristics of an environment and error. Some of these necessary relations have been researched in isolation, but often not to the degree necessary or combined to form a complete picture of the process. The areas researched fall into four major categories, analysis of: propagation, algorithms, environments and system opimization.

Propagation Analysis

The first source of error in any laterative localization system is determining the distance some signal has traveled. While a signal's power will decay solely due to distance covered while propagating, it can also incur losses by interacting with the environment. While it is quite easy to model power loss due solely to distance propagated with a simple lognormal relation, it is a different matter to model the influence of all distorting elements. Being able to determine the exact model and parameters to use would require extraordinary knowledge of the signal environment.

Different models are tuned to model different types of distortion interactions based on a particular system's design. An indoor environment is quite distinct from an outdoor one. Indoor environments have a large number of reflective, absorptive and diffracting objects within them. To counter this, indoor communication systems often have very short wavelengths so that signals are prone to shatter and reflect when encountering an object, forming an incidental path by filling the entire space with energy, increasing the likelihood that the signal will encounter multiple distortions. [39] There are a plethora of propagation models used for mobile and indoor systems such as the Bullington model, the model of Okumura et a1., the lTV (CCIR) model, the Hata model, the Ibrahim-Parsons model, the Joint Radio Committee(JRC) model, the Ikegami method, to mention only a few. [3, 2] In practice the simple lognormal model for power loss over distance is used in some manner in the overwhelming majority of indoor laterative localization systems. [2, 16, 9, 40, 34, 10]

One popular method of determining parameters to the lognormal model is to assess the distortive strength of different building materials, sum their effects over a straight line path from the transmitter to receiver, and then solve for parameters that would correct for those objects' influences. [2] While this is a workable first-order approximation it requires a detailed environmental survey. The International Telecommunications Union generalized this parameterization process by assessing the likely parameter sets for a series of generic indoor scenarios and averaging them to establish an expected parameter range given an environment's general disposition. [33] Other methods employ machine learning, statistical analysis, averaging or other such amoritizing mechanisms [16, 34, 40, 14]

One of the fundamental difficulties of such parameter estimation methods is predicting or analyzing error. Using one set of parameters to represent the plurality of all possible paths that a signal may take to a given LM already invites significant error. [7] Methods of determining distance estimation error differ by algorithm and application. In cases of radio tomography, only detecting a signal power different enough than an expected value is enough to indicate an event. [27, 15] While many analyses of distance estimation error for laterative localization algorithms have been done, they are often too far removed from actual data to yield actionable results. An error metric often used is the Cramer-Rao Lower Bound. It however requires continuous relations to compare against, which can not be obtained by measuring at discrete locations in an actual environments. In order to analyze distance prediction error using such common techniques researchers will often either alter the metric, amortize the data, use simulated data or use a related but entirely different ranging modality, like time of arrival. [41, 6, 8, 5] The fundamental difficulties in assessing distance estimation error are having a metric that: can be applied to discrete signal data sampled from a physical environment and clearly determines the degree to which the signal data differs from the lognormal model in such a way as to estimate the likely resulting distance misestimation.

Algorithm Analysis

The second source of error in laterative localization systems is the localization algorithm itself. Ever since the widely-regarded RADAR paper in 2000, many different studies have been done to determine how to improve the accuracy of 802.11 signal strength-based localization systems. [9] RADAR is a scene-matching algorithm that uses a nearest-neighbor strategy to assign locations to RSS fingerprints based on its training data. Training interpolation, environment gridding and tiling were later introduced as improvements over RADAR in the SPM and Rice University's algorithms. [16, 19]. In order to increase localization accuracy, probability-matching algorithms were engineered. These algorithms still match testing fingerprints against a training set, however the matching mechanics are based on probabilistic mechanisms rather than a more direct comparison of signal strengths. Such techniques are more resilient to the ever-present random signal fluctuations and pertubations of indoor environments. The Nibble algorithm was one of the first to implement probability-matching, using a neural net to match fingerprints to locations. [12] The ABP algorithm and the HORUS system built on this design. [16]

Fully probabilistic algorithms followed. These algorithms still use some training data, but need significantly less than matching-type algorithms before their estimators are saturated. Probabilistic algorithms directly compute coordinates for a fingerprint probabilistically rather than by selecting a set of coordinates from a pre-measured list. Algorithms such as M1 and M2 directly compute a fingerprint's likely coordinates after generating a series of attenuation and bias corrections to a lognormal signal model based on training data and using it to translate testing data into ranges to laterate on. [16, 24] While these and other algorithms have different methods of computation, error performance is often strikingly similar when tested in the same environment, with no clear improvement. [16]

While all reasonable attempt at improvement based on rational expectations or statistical arguments, localization algorithms can only be improved heuristically at best. The difficulty in engineering localization algorithms is that there is no clear relation between a set of distance measures and positioning error. Since the error context in which the algorithm is computing can not be exactly known, the only reasonable solution is to attempt to amortize the effect of error, be it environment gridding, averaging or statistical solving. These mechanisms act appropriately, but serve to further muddy the waters by blending good data with bad to raise the localization error floor. Some attempts at dealing with this problem involve constraining the range of possible solutions by using only connection-oriented localization. Connection-oriented localization determines that if one node can see another, they must be closer than some maximal distance, reducing the localization model to a signal power threshold based on radial range. [11] While this does reduce the complexity of the model, it hides the very effects of the environment that caused the error that may provide useful clues. Rather that direct thresholding, one improvement is to base detecting-oriented ranging on maximal relative difference between subsequent signals as a measure of link quality. [14] Such models however lose the ability to position absolutely and can only determine the likely relative distance between nodes. Another improvement is to add additional information by instrumenting link quality along with signal strenth. [35] Such an alteration is far from exact since link quality is an uncertain metric that can often be an informational measure of decoding error, and may rely on noise power, chip rate and modulation rather than only signal strength. [35]

Environmental Analysis

Compared to algorithmic and propagation analysis, environmental studies are often very narrow, particular to a given environment or algorithm. The environment is often investigated only tangentially while evaluating the general disposition of an area and the average efficacy of the lognormal model. [20, 7] In order to auto-correct for lognormal model error, some studies have estimated common estimation error scenarios and have prepared basic heuristics to smooth over them. In so doing however they have necessarily generated an expected environmental distortion map and identified distortion scenarios, but did not not take the conceptual step to generalize from error in the lognormal model to a parameterization of signal environment itself. [32] Other studies have followed the ITU model and have computed lognormal propagation parameters over certain environment areas, but rather than defining their areas based on floorplan data, they define their areas as concentric ranges from LMs within a variability threshold. [31] Such a construction is all but a defacto general distortion model of the environment itself, but is put forward solely as a lognormal model parameter estimation aid.

Systems based around radio tomography often track closely to the ITU environmental model, assessing radio behavior in a few different environments and averaging results. Some such studies have demonstrated useful general parameters like optimal antenna separation to detect a break in a link or the capability to recognize a particular slice of the signal environment. [1, 21] While these are not conclusions on the disposition of distortion in an environment, they are useful insights into the expected amount of deviation per square foot of monitored area; a volumetric estimation of environmental noise. Similar work has been even more exact, automatically assessing parameters to the lognormal model and estimating error area, but not generalizing these mechanisms to draw conclusions about the environment at large. [15, 27]

While all informative, such studies define the problem much too narrowly. Most consider only bounding or improving accuracy for a given algorithm, or determining propagation parameters for a given environment. Few have attempted to form a systematic basis of comparison in order to understand the entire problem, as it would require a solution for any environment, algorithm and distortion type simultaneously. We propose just such a solution. We will present a method to detect systemic distortions common to radio indoors in the 802.11 band, a method to benchmark algorithm performance in the presence of these distortions and a method to decompose an environment into a series of distortion profiles. By so doing we will specify how anyone can construct a synthetic error model for any environment, algorithm and distortion suite that is evaluative, analytic and diagnostic.

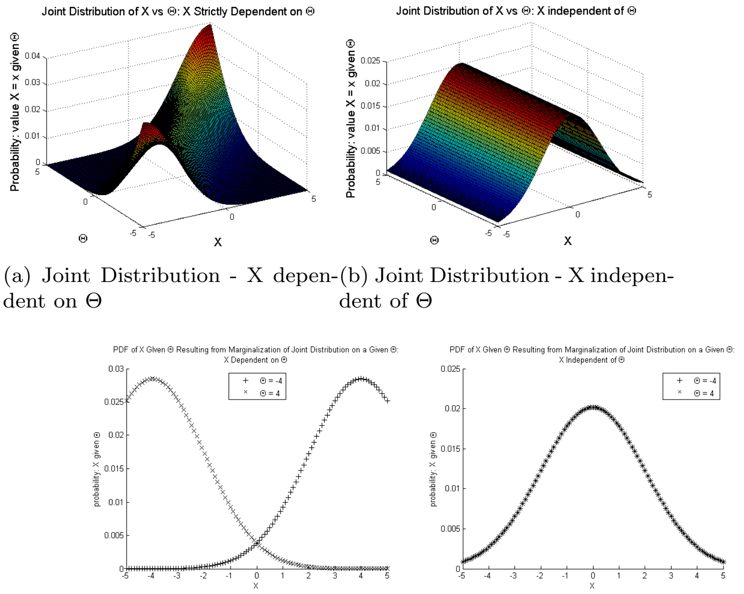
Chapter 3
Quantifying Ranging Error

Signal power is conditionally dependent on the distance the signal traveled. If there were no distortions, the opposite would be true as well. While it is impossible to know all the effects that have contributed to a specific RSSI, the propagation mechanics of radio are thoroughly studied, and several classes of distortion have been identified. Laterative localization algorithms compute based on the assumption that their signal-to-distance model can correctly convert RSSIs to ranges. If all ranges are free of distortions and errors, the lateration stage should result in the exact coordinates of the transmitter. Bounding, analyzing and understanding deviation in signal to distance translation is tantamount to determining the expected degree of laterative localization algorithm performance. While we do not claim to be able to deduce all the subtle effects acting on a single signal, we assert it is quite eminently possible to determine the single dominant distortion type, how strong it is, and how much of the signal it affects.

3.1 Distortion Detection Metric

When calculating ranging error it is a simple matter to correlate RSSI with the true, recorded distance to a sample point. Such a measurement is compelling due to its straightforward examination of the lognormal path model, however it forces the assumption that the lognormal model represents the sum total of all influences on RSSI. Given that the lognormal model is parameterized by several values, it is preferable to instead regard estimation of the parameters themselves as the source of the error and to treat the difference between the lognormal model that would result from a given parameterization and the physical environment as itself the error model. The difficulty with such a construction is determining a method to measure this difference. The Cramer-Rao Lower Bound (CRLB) is often used to assess how well a model fits a given process and is often applied to localization systems. We will demonstrate that the CRLB is in fact fundamentally unfit to analyze this data and present a similar novel metric that can compare models, and/or a discrete sets of data in order to detect and classify well-known types of systemic radio propagation distortions.

3.1.1 The Cramer-Rao Lower Bound



(a) Joint Distribution - X dependent on Θ

(b) Joint Distribution - X independent of Θ

(c) Marginalized Dependent Distribution:
PDFs on X with Θ = 4 and -4

(d) Marginalized Independent Distribution:
PDFs on X with Θ = 4 and -4

Figure 3.1: Probability Spaces of Variables with Varying Conditial Dependance

The Cramer-Rao Lower Bound (CRLB) is a method that determines the minimal achievable standard deviation of an unbiased estimator of a random variable. It functions by determining the expected rate of change in the likelihood function of the random variable's estimator conditioned on a value of the random variable. The central argument of the CRLB is that if some parameter, Θ is a good estimator of X, then by definition selecting a certain Θ will result in a probability distribution on X with a very low standard deviation. X and Θ can both vary though, so if X is strongly conditioned on Θ then each Θ should result in a very different distribution on potential values of X, as is illustrated in figures 3.1c and 3.1d below for propositional distributions in which there is strong and no conditional dependence of X on Θ, respectively. Instead of fixing Θ and varying X to generate a distribution on X, fixing X and varying Θ would result in a likelihood function on values of Θ. The resulting likelihood function on Θ should have a clearly dominant value, the value that would result in the distribution X with the highest likelihood of that given x = X occurring.

The Cramer-Rao Lower Bound is defined by computing derivatives across the space of expected values. These derivatives impose necessary regularity conditions that make them particularly unsuitable for use with sampled radio data. It is patently not possible to sample signal power at an infinite number of discrete physical locations. Even if it were possible, distortions are applied when the signal encounters an object at random places in the environment, so any signal magnitudes sampled at regular distance intervals, describing a signal/distance vector, will have jump discontinuities in it wherever an object was encountered, causing the resulting data to be undifferentiable. Due to the non-continuous and discrete distortive nature of an actual environment, the CRLB can not be used without modification. One possible such modification is to determine a differentible function that approximately describes the signal data. [41, 35] Such solutions strongly beg the modeling question further.

Approximating signal data with some function in order to make it comparable using the CRLB covers up the discontinuities caused by the irregular propagation environment. Such approximations in effect remove the very aspects of the signal data that make it difficult to fit to the lognormal model and cause ranging misestimation. Such approaches also require an estimate of how well the approximating function itself fits the sampled data, a problem that is reducible to the original problem of determining how well the lognormal model fits the sampled data in the first place. Such approaches, while of academic interest, still do little to determine how well the lognormal model describes a set of discrete data samples. We assert that in order for any ranging error estimator to be feasible, it must not obscure sampled data, or must at least be parameterizable to such a degree that it can compare sampled data directly to a model. If the data is not complete enough to supply the metric, then the rigor of the metric needs to be relaxed. We find the CRLB a compelling metric because it compresses a large range of measurements to a single probabilistic indicator that is related directly to a proposed model, or a particular parameterization of the model.

3.1.2 The Discretely Distributed Log Hölder Metric

While the exact mechanics of a radio environment can not be known precisely nor be tested exhaustively, it is necessary to be more sparing in what is evaluated at the cost of a less precise result. Sampling RSSIs at regular intervals from a single straightline path provides a snapshot of the effect of the radio environment at each step. Since a signal-to-distance model defines the expected sensitivity of RSSI to distance, the first goal is to assess the signal-to-distance sensitivity from the actual environment. Once that is computed, we next have to convert the assessment into a form that is readily comparable to the suggested lognormal model and compress the result to a single parameter, so that multiple propositional lognormal models can be evaluated. We define a method similar to the CRLB by computing the discrete distribution of the logarithm of the Hölder Metric (DDLHM) of a sequence of adjacent signal strengths sampled at regular distance intervals from a landmark.

Computing the DDLHM

Let d_i be the distance from the landmark at point i

Let RSS_i be the ith scalar signal power magnitude, sampled at range step i from a given landmark

Compute the Hölder Metric between each adjacent pair of scalar signal powers: $HM_i = \left\| \frac{RSS_{i+1} - RSS_i}{d_{i+1} - d_i} \right\|_2$

- The Hölder Metric is used in this case for completeness and universality of the metric's definition. In this case however, even though the Hölder Metric does indeed include a matrix norm, it is computed on a scalar quantity and results in a scalar quantity.

Compute the logarithm of each Hölder Metric: $LHM_i = log_e(HM_i)$

Compute the discrete distribution of the Log-Hölder Metrics for the vector being analyzed: $DDLHM = DD\left(LHM\Big|_1^n\right)$

The resulting discrete distribution tabulates the sensitivities of RSSI to distance traveled at a series of steps through a signal environment. Evaluating a proposed lognormal model at the same distances and applying the same process would result in a similar discrete distribution. The degree of similarity between the two discrete distributions determines the degree of similarity between the predicted RSSI to distance sensitivity predicted by the model and experienced in the environment.

Determining Model Distance with the DDLHM

The DDLHM results in a discrete distribution and not a single comparative value. In order to judge distribution similarity, we compute the Jensen-Shannon Distance (JSd) between the DDLHM of two signal sample sets. The JSd is the average of the Kulbeck-Leibler Divergence (KLd) between each of the distributions and their average.

x**:** A value that can occur

$A(x)$**:** Probability of value x occurring in distribution A

$B(x)$**:** Probability of value x occurring in distribution B

N**:** Number of distributions being compared, equal to 2 below

$KLd(A||B)$**:** Kullback-Leibler Divergence of distribution B from A

$$KLd(A||B) = \sum_{i=i}^{n} \left(\frac{A(x_i)}{B(x_i)} \right) * A(x_i) \tag{3.1}$$

Although a popular and well-defined metric, the KLd is fundamentally unfit to measure the differences between discrete distributions of samples from a real environment or propositional, possibly incomplete, distributions or models. The KLd requires that the histograms compared be defined over the same range of values. If either histogram compared has no (0) occurrences of a certain value where the other does not, the KLd will collapse to Infinity or Not-a-number. All the distributions compared here are expressly expected to contain discontinuities and distortions and must still be comparable to distributions calculated directly from a proposed lognormal model. A technique used to compensate is to substitute a 0 for the KLd of degenerate outcomes, however this technique discards large amounts of data and results in incongruous measures.

The KLd is also directional. It determines the bits of difference of one distribution from another, not the bits of difference between both. This directionality works well if one distribution is known to be exact and the other is known to cover at most only a subset of the total possible values. Our analysis requires that we compare multiple hypothetical distributions to live data, making it impossible to have a single fundamental distribution to compare against. The very issue we address is that we can not know the precise signal-to-distance model.

The Jenson-Shannon Distance (JSd) solves these difficulties at the cost of a less intuitive measure. JSd computes the average of the KLd between each distribution and the average of all distributions being measured. This solves the degenerate distribution problem since each distribution is compared to the average of all distributions being measured, which includes itself, so zeros are contained in the numerator. The JSd determines the informational distance between all the distributions compared, not the divergence of a specific one from another. The JSd's measurement is a normalized ratiometric informational distance between the distributions measured.

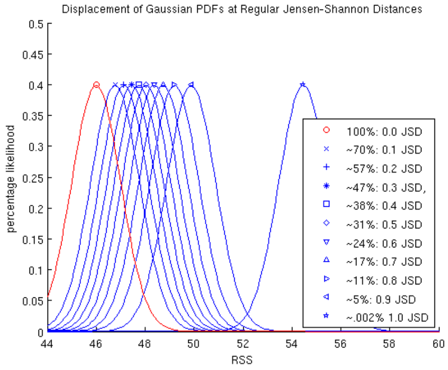


Figure 3.2: JSds of Gaussian PDFs only varying mean

$JSd(A||B)$**:** Jensen-Shannon Distance between A and B

$$M = \left(\frac{A+B}{N}\right) \tag{3.2}$$

$$JSd(A||B) = (.5 * KLd(A||M)) + (.5 * KLd(B||M)) \tag{3.3}$$

The JSd varies from 0 exactly to 1 asymptotically. Since JSd is computed as the difference between a distribution and its average, the JSd between any distribution and null would be 0.5, so any reasonable match should be below 0.5 at minimum. Figure 3.2 depicts the JSD between a Gaussian PDF with standard deviation 1 and mean 0 and several others whose means have been shifted enough to have the JSd between the two to settle on tenth-JSD increments. For our purposes, we'll define a good match between signal-to-distance DDLHMs as 0.25 JSd or less, a very good match as 0.2 JSd or less and an excellent match as 0.15 JSd or less.

The DDLHM and the JSd provide us with the desired characteristics of the CRLB, but without a reliance on continuous and differentiable functions that can describe our data. We first need only sample RSSI values from our environment over a straight line at fixed distance intervals and compute the DDLHM to summarize the actual RSSI to distance senitivities. We can then generate any number of possible lognormal models with different parameters, compute the expected RSSI values at the same distances sampled in the physical environment, and compute the DDLHM of that data. It is then a simple matter to compute the JSd between the DDLHM of the physical data and the DDLHM of each hypotetical lognormal model to determine which parameterization matches best. If we carefully choose parameters to the hypothetical lognormal models to reflect the measured behavior of common indoor distortions, we can conclude that the parameters that resulted in the best match most accurately describe the dominant type of distortion along that sampled path in the environment. What is needed is a reasoned and measured approach to generate such parameters to the lognormal model.

3.2 Distortion Types

Nearly all laterative localization algorithms use the lognormal path-loss model: $a * -10 * log(\frac{d}{d_0}) + b$. [17, 18, 22, 7, 9, 16, 3] The 'a' and 'b' parameters are referred to as the 'attenuative' and 'bias' parameters, respectively. Any algorithm that uses the lognormal model must decide upon a value for these parameters, as they govern the behavior of the model. Computing the lognormal path-loss model over a series of distances with an 'a' value of 1 and a 'b' value of 0 results in RSSI values free of distortion, or 'freespace' values, dipicted in Figure 3.3a. Varying 'a' and 'b' results in the first two systemic distortion types, attenuation and bias, respectively. Bias is a RSSI amount that shifts the output of the model by some set amount.

Beyond attenuation and bias, which are directly addressed as lognormal model parameters, we also consider multipath propagation. The indoor environment consists of many highly radio-reflective elements like glass, structural elements, ductwork, and metal shelving. If two strong signal components destructively interfere, they can cancel each other. The result is very large drop in signal power at a specific point even with no apparent obstruction. Figure 3.3e depicts such a multipath scenario. We only consider two multipaths with the interfering component bouncing at a point equidistant between the transmitter and receiver.

Distortion Parameters

Beyond the type of distortion, we identify two other qualities to detect: the strength of the distortion and the range at which it started affecting the signal. Distortion strength ranges differ per distortion type. The bias distortion type is a single additive parameter to the lognormal path-loss function as it models signal loss due to a single, largely opaque object. No bias, or a bias of 0 dBm, is the result of perfect, unobstructed propagation. Being that only integer values are reported, the minimal bias value is 1. We select 30 as a maximal value since the entire parameter space spans from approximately -40 dBm at 1 meter to -99 dBm, beyond which signal gains are due to coding only. [39] Given that the entire parameter range is nearly 60 dBm, we take half of that as a maximal deviation from the expected value. A drop of 30 dBm, or half the entire parameter space, is quite a striking drop and well beyond what should need to be tested for difficulty of detection. Since the vast majority of RSSIs are reported as whole numbers, we will consider integer dBm values only. We consider the valid range of bias parameter powers to be an integer between -1 and -30, for 30 possible strength values.

The attenuation distortion type is a single multiplicative parameter to the lognormal path-loss function as it models the effect caused by a series of small obstructions or propagation through a single, large, semi-permeable medium that reduces signal power gradually over distance. An attenuation parameter of 1 represents absolutely no additional attenuation, and accounts only for power loss due to propagation over distance. In their analysis of attenuation parameters caused by common building materials, the ITU recommends an attenuation parameter between 2.5 and 3, with the maximum being 5. [33] Many localization systems tend to pick a value within this range. [36, 20, 23, 3, 31] We consider values from 1.05 to 3.5 in steps of .05, for 50 possible strength values.

Since multipath is a quality of propagation that is not accounted for in the common lognormal model, we calculate it separately and compute a weighted average between the powers applied to the two paths. No multipath, or 0%, is only the lognormal model computed with no obstructions (attenuation 1 and bias 0). Full multipath, or 100%, splits the power equally between each path, with half the power going to the direct path (lognormal propagation), and half the power going to the multipath. We consider percentage amounts of multipath ranging from 2.5 up to 100 in steps of 2.5 for 40 possible strength values.

The range at which a distortion starts affecting a signal is quite important since, in the absence of such a measurement, the effect of a distortion is averaged over the entire length of the path of propagation, rather than being applied only to the point at which it began affecting the signal. Figure 3.3b depicts attenuation at different strengths over 20 meters, while Figure 3.3c depicts attenuation under the same conditions, bit only after encountering some object after 10 meters.

For our initial exhaustive tests we sample at half the wavelength in order to be sure of detecting all distortions on order of the size of the wavelength and consider all distances ranging from 1 meter from the transmitter to 20 meters in steps of half the wavelength of 802.11 channel 6, or .0615 meters, for 309 possible incident distance values. [39]

In order to detect distortion type and parameters, all that is necessary is to generate a hypothetical version of the lognormal model for each distortion type and parameter to be considered, apply the DDLHM to reduce it to a sensitivity distribution and to compare that distribution to the DDLHM of RSSI samples from the physical environment using the JSd. Each hypothetical set of parameters would result in some JSd value. While this allows the fitness of the different parameters to be measured, it does so at the cost of a less apparently analyzable metric. Below we will investigate the operating tolerances of the DDLHM and JSd on an exhaustive and complete range of propagation parameters. While there may be more subtle types of radio distortion or more than one type of distortion applied to the same group of RSSI samples, we only attempt to identify the single dominant distortion type from the range we've defined.

3.3 Properties of the Exhaustive Distortion Parameter Set

3.3.1 Exhaustive Distortion Parameter Set

Above we have defined how one can translate the degree of similarity of a certain parameterization of the lognormal model to data sampled from the physical environment into a DDLHM JSd value. Based on the definition of the JSd, a smaller value indicates a better match, so determining the fitness of a single set of parameters is straightforward. We however intend to determine the parameters that best describe the dominant distortion over a set of samples by examining the DDLHM JSd between those samples and propositional samples computed using an exhaustive set of parameters for each distortion type. The range of possible distortion powers and incident ranges for each of the three distortion types identified above describe a state space of 37080 different possible parameterizations: (30 bias strengths * 309 incident distances)+(40 multipath strengths * 309 incident distances) + (50 attenuation strengths * 309 incident distances). While a single DDLHM JSd's meaning may be apparent, it may not be apparent in what context to judge an ensemble of 37080 DDLHM JSds resulting from an exhaustive analysis.

3.3.2 Baseline DDLHM JSd behavior

Attenuation

It would be reasonable to presume that JSd would increase as distortion strength increases and as incident distance decreases. Encountering an attenuating obstacle earlier in the path of propagation rather later also increases the DDLHM JSd, for the most part. As can be seen in Figure 3.4 attenuation factors from 1.25 to 1.5 result in a smaller DDLHM JSd value if they are incurred before 8 meters from the transmitter, steadily increasing and peaking at 8 meters, after which the values begin to decrease. Since the DDLHM JSd measures the overall number of differences in signal strength to distance sensitivities over two sets of samples and not absolute difference in signal power. Certain configuration of parameters can conceivably result in a lower DDLHM JSd even if they indicate a stronger distortion. Early in the first few meters of propagation signal power drops drastically per unit distance. Some distortion parameterizations can more closely mirror this natural signal decay than others, even if, and especially, they result in stronger distortions than others.

The relationship between incident distance of a distortion and DDLHM JSd value is also not linear, but varies less than the attenuation strength value demarcating ranges of JSd. As can be seen in Figure 3.7, no attenuation that is applied after 16 meters is particularly detectable, no matter how strong it is. At that point the 802.11 signal power is so low the attenuation effect is significantly diminished. Another general rule that can be extracted is that only very strong attenuations (above 2.5) are detectable beyond 10 meters. Due to the JSd depression effect for extremely dissimilar distributions, no attenuating obstacle that is encountered beyond 10 meters will result in a JSd over 3.75. These peculiarities together mean that most detectable attenuations will have a strength parameter between 1.25 and 3 and will occur within the first 8 to 10 meters.

Bias

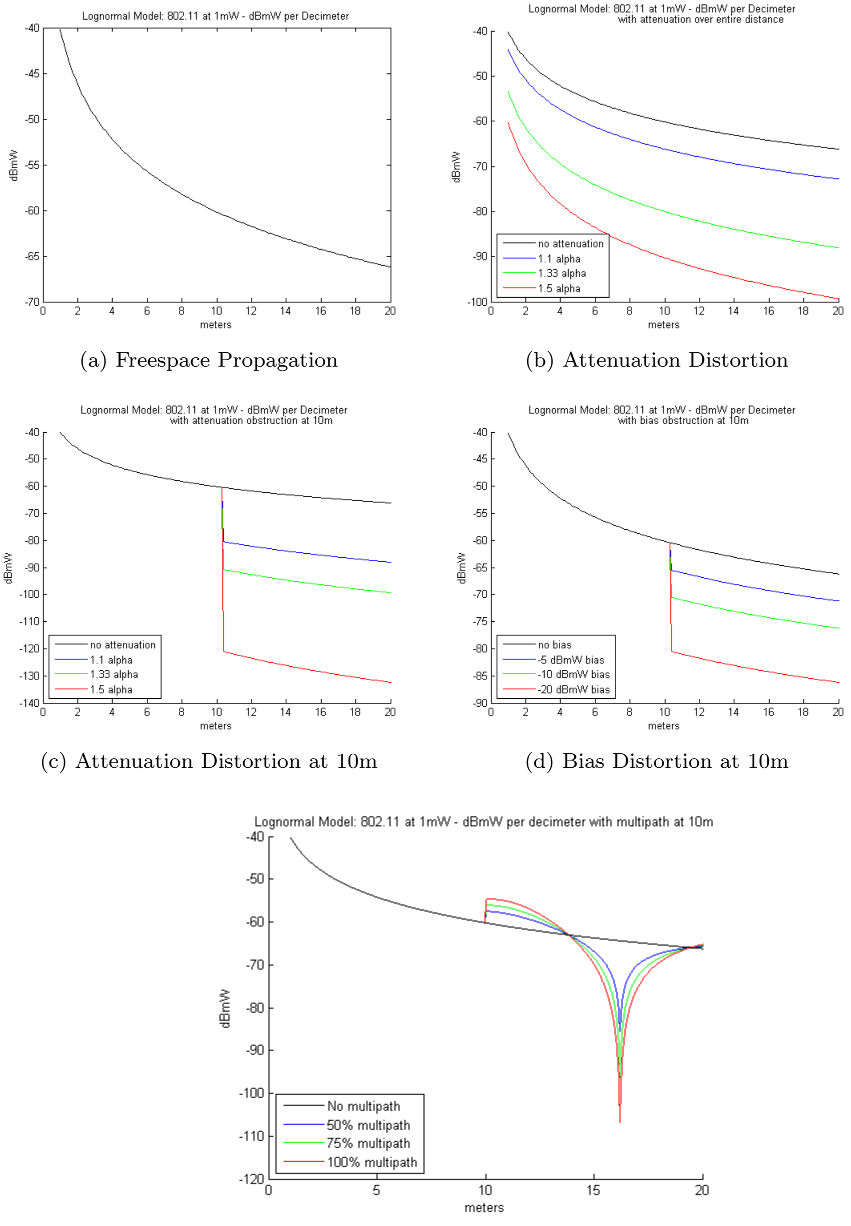
The aliasing between distortion strength and JSd can be seen more sharply in bias distortions. In Figure 3.9 a -15 dBm drop can result in either a DDLHM JSd of 0.4 up to 0.65 inside of 7 meters. Bias is a one-time application of a shift in the signal power, so it is often immune to the extending sensitivity JSd aliasing of attenuation. This displays another effect of the DDLHM JSd. This shift can quickly push the resulting discrete distribution well past the range of the unobstructed lognormal's DDLHM. Since we do not clip any elements of the discrete distributions, the degree of distortion for high values of bias cause relatively few samples to fall into the same discrete distribution bin. The JSd metric often results in large differences when comparing discrete distributions with very different values for the same range, but when comparing any number of values in one distribution against a bin that the other distribution has none in, the JSd is capped at 0.5, resulting in a depression of JSD values the more out of phase two discrete distributions are. Incident distance is more directly diagnostic to bias JSd rather than attenuation due to this shifting effect, as can be seen in Figure 3.11.

Bias distortions behave in a manner similar to attenuation in other respects. When the bias occurs fairly close to transmitter it may be somewhat difficult to detect. Since the DDLHM determines the point to point change in signal-to-distance sensitivity and the signal power drops very, very quickly over the first few meters a strong bias distortion may cause a change in signal sensitivity very similar to unobstructed propagation at short ranges as can be seen in Figure 3.12. Even though bias distortions with a signal drop of -15 dBm may be harder to directly inside of 9 meters, the DDLHM JSd is at or exceeds 0.5 over the entire range, indicating an extremely bad match at all points.

Multipath

Multipath distortions are considerably more noisy than bias or attenuation types as they fluctuate strongly throughout their propagation, from increasing signal power by a factor of 4 to reducing down to nearly 0. As decidable as the DDLHM JSd of attenuation and bias distortions are based on distortion strength and incident distance, multipath is distinctly less so. Even though each multipath profile is very noisy and is quite unlike unobstructed lognormal, they are all unlike it in the same manner. This can cause multipath DDLHM JSds to be very similar. While bias and attenuation can cause maximal DDLHM JSd values above 0.9, multipath distortions maximize at just under 0.25, as seen in the scale of Figure 3.14.

Multipath distortions behave quite differently than bias and attenuation. The most disruptive effects of multipath propagation are the occasional nulls that can occur due to signal cancellation. Somewhat counterintuitively, a small multipath component can skew the distribution of signal-to-distance sensitivities more than a very strong multipath component since stronger multipath components result in deeper cancellations. The larger number of deeper cancellations begin to look proportionally similar to common lognormal propagation, where there are very fast rates of changes of sensitivity over the first few meters. This can be seen in Figure 3.14, multipath scenarios where the multipath component is less than 40% the strength of the direct path component causes maximal dispersion of the DDLHM. Another difference between multipath and the other two distortion types are the characterizing effect of these nulls. Nearly any multipath encountered before propagating 12 meters causes a DDLHM JSd of at least 0.14, while nearly all encountered afterward do not.



(a) Freespace Propagation

(b) Attenuation Distortion

(c) Attenuation Distortion at 10m

(d) Bias Distortion at 10m

(e) Multipath Distortion at 10m

Figure 3.3: Systemic Radio Distortion Types

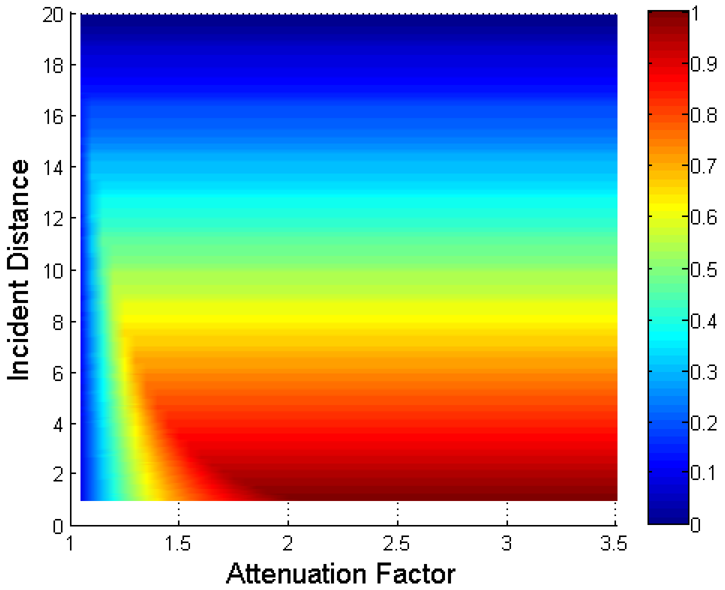


Figure 3.4: DDLHM JSd between Exhaustively Attenuated Lognormal and Unobstructed Lognormal

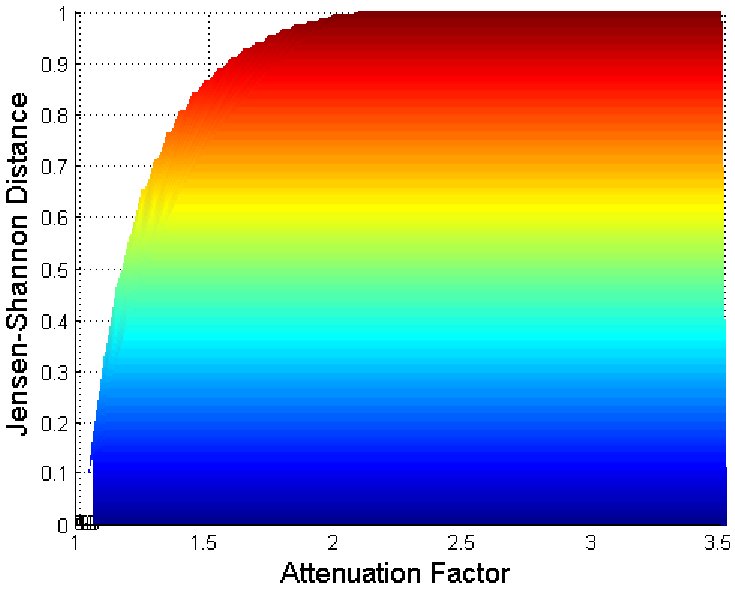


Figure 3.5: DDLHM JSd between Exhaustively Attenuated Lognormal and Unobstructed Lognormal: Marginalized on Distortion Strength

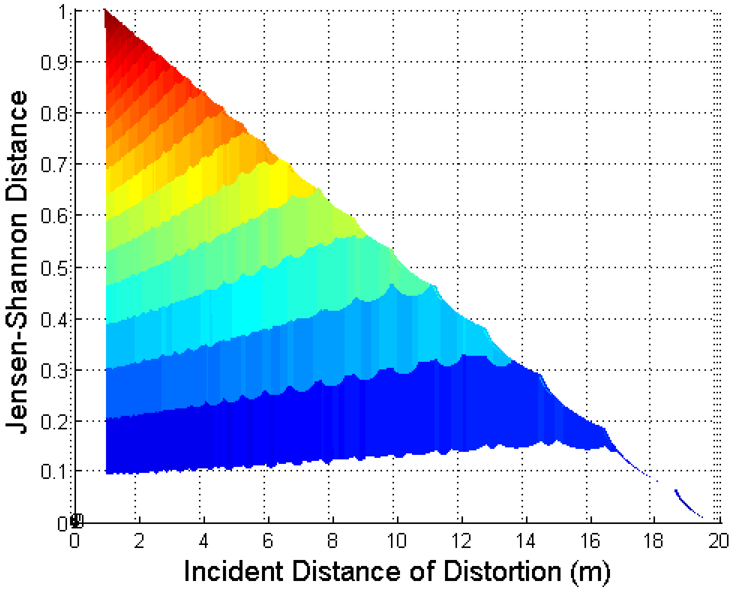


Figure 3.6: DDLHM JSd between Exhaustively Attenuated Lognormal and Unobstructed Lognormal: Marginalized on Incident Distance

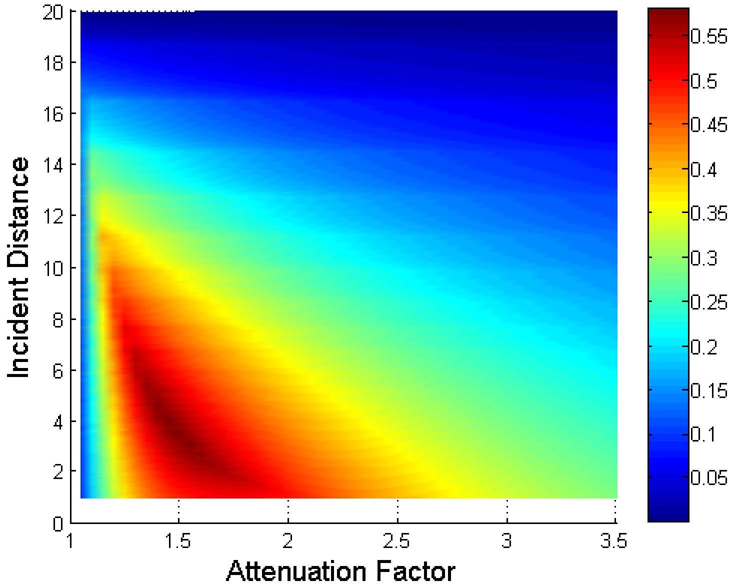


Figure 3.7: DDLHM JSd between Exhaustively Attenuated Lognormal and Unobstructed Lognormal: JSd per Attenuation Parameter

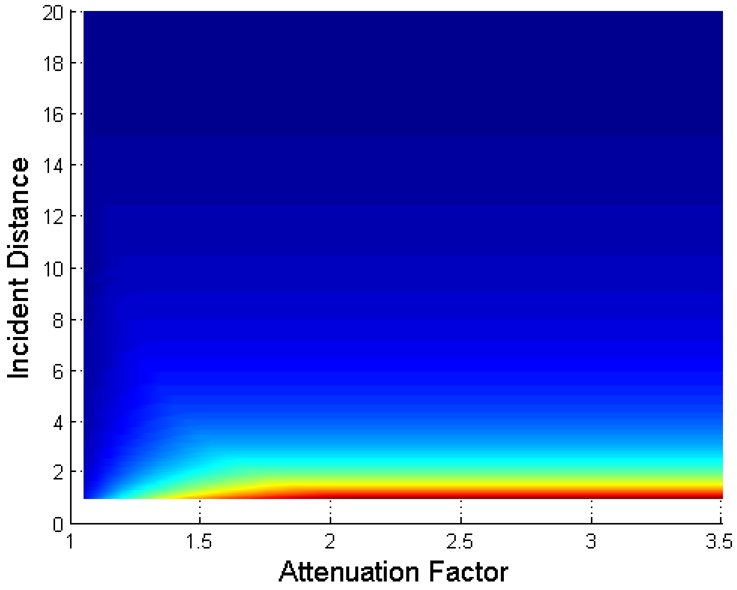


Figure 3.8: DDLHM JSd between Exhaustively Attenuated Lognormal and Unobstructed Lognormal: JSd per Incident Distance

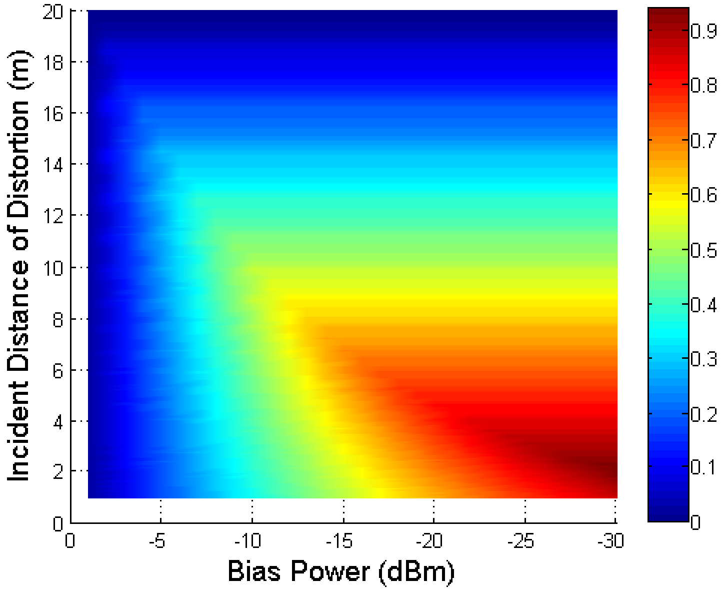


Figure 3.9: DDLHM JSd between Exhaustively Biased Lognormal and Unobstructed Lognormal

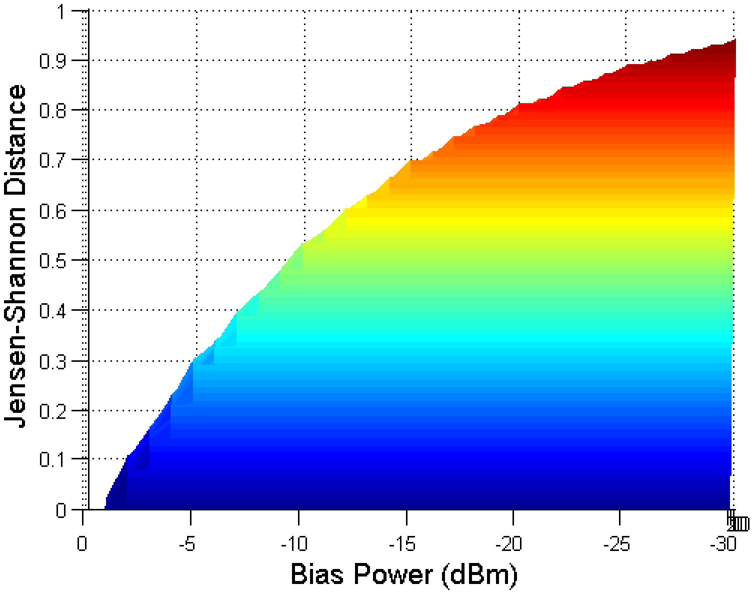


Figure 3.10: DDLHM JSd between Exhaustively Biased Lognormal and Unobstructed Lognormal: Marginalized on Distortion Strength

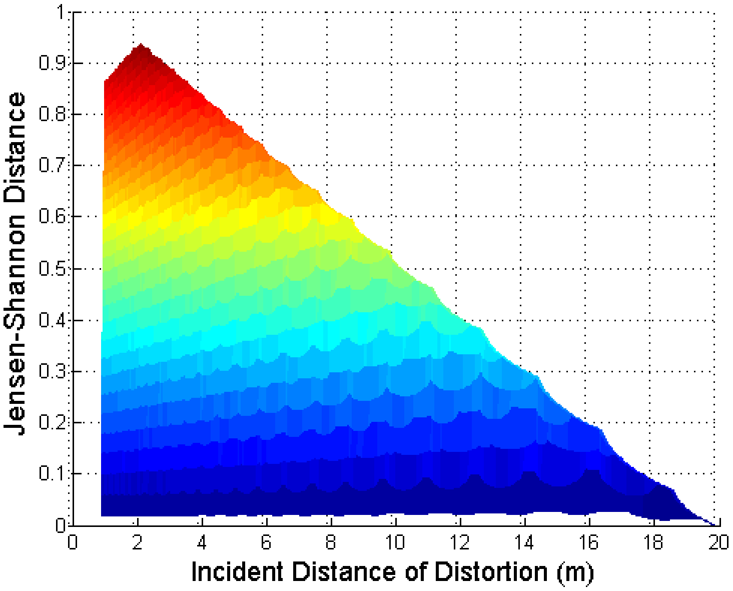


Figure 3.11: DDLHM JSd between Exhaustively Biased Lognormal and Unobstructed Lognormal: Marginalized on Incident Distance

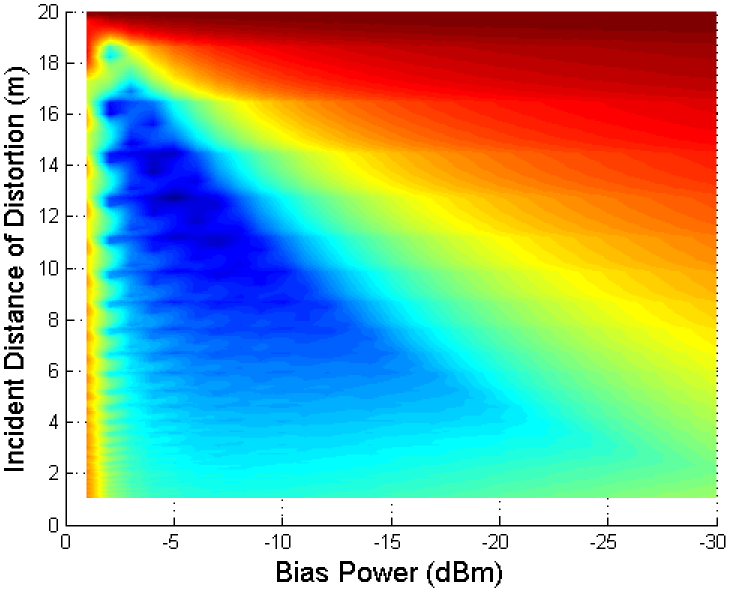


Figure 3.12: DDLHM JSd between Exhaustively Biased Lognormal and Unobstructed Lognormal: JSd per Attenuation Parameter

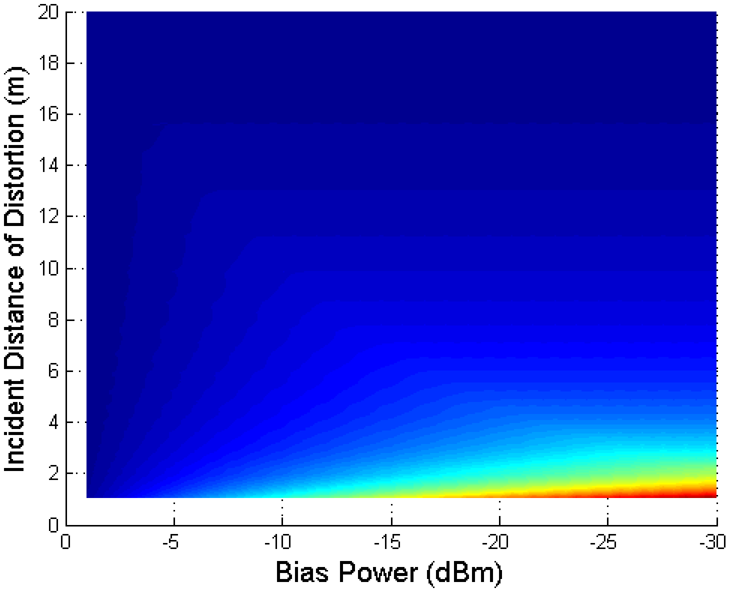


Figure 3.13: DDLHM JSd between Exhaustively Biased Lognormal and Unobstructed Lognormal: JSd per Incident Distance

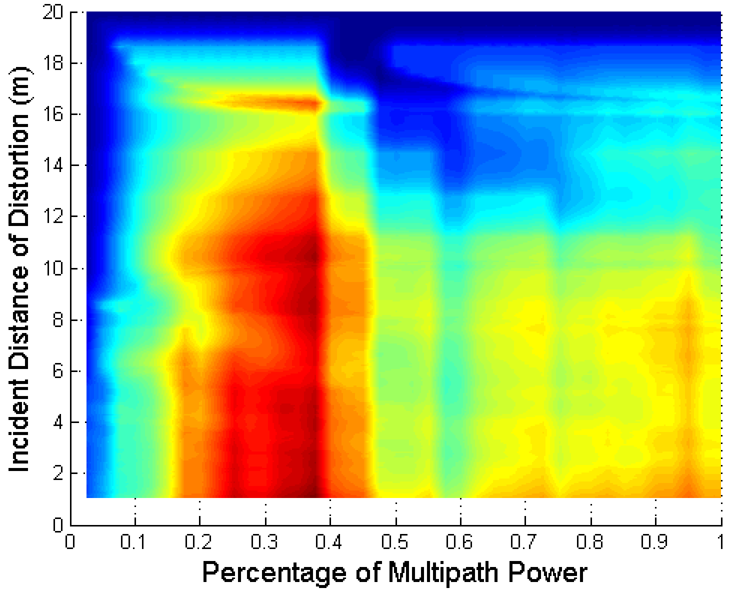


Figure 3.14: DDLHM JSd between Exhaustively Multipathed Lognormal and Unobstructed Lognormal

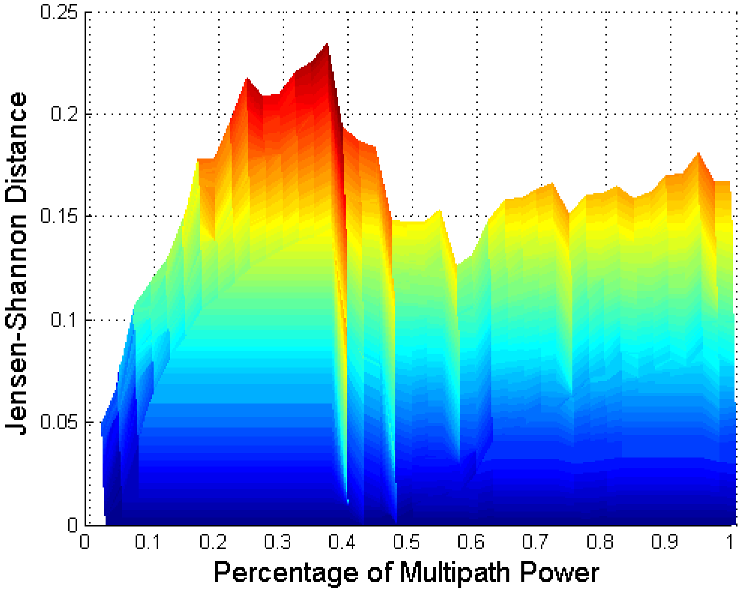


Figure 3.15: DDLHM JSd between Exhaustively Multipathed Lognormal and Unobstructed Lognormal: Marginalized on Distortion Strength

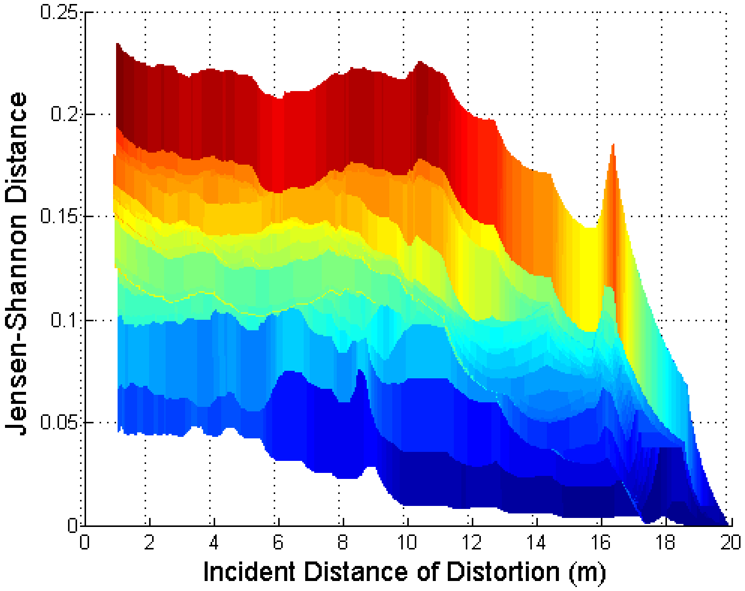


Figure 3.16: DDLHM JSd between Exhaustively Multipathed Lognormal and Unobstructed Lognormal: Marginalized on Incident Distance

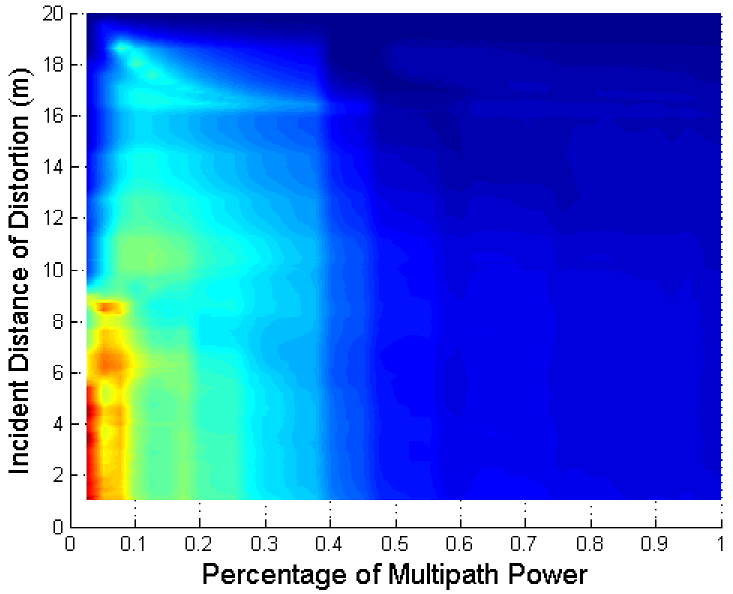


Figure 3.17: DDLHM JSd between Exhaustively Multipathed Lognormal and Unobstructed Lognormal: JSd per Attenuation Parameter

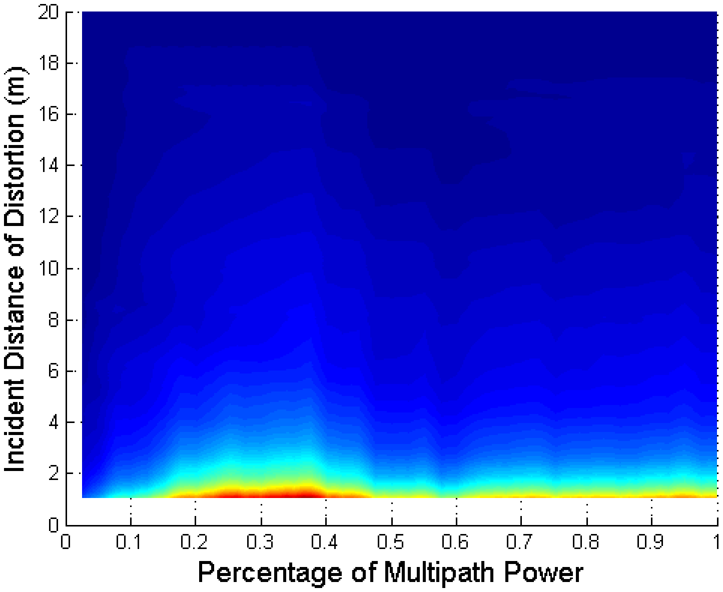


Figure 3.18: DDLHM JSd between Exhaustively Multipathed Lognormal and Unobstructed Lognormal: JSd per Incident Distance

Limits of the Exhaustive Parameter Set

In order to determine the significance of the DDLHM JSd between two different parameterizations it is necessary to know how sensitive the DDLHM JSd is to changes in the parameter values. Since we have enumerated the entire parameter space, it is possible to exhaustively compute the DDLHM JSd between each of the 37080 different parameterizations of each distortion type to every parameterization of all other types. Analysis of the behavior of the DDLHM JSd will determine how well the process can discriminate between different instances of the same distortion, directly informing the construction of a confidence interval to be applied to future measurements. The computational cost of such an exercise is infeasible. Since the JSd is symmetric, two parameterizations need only be computed once. The attenuation feature space, for instance, consists of 15450 distinct distortion power, incident distance pairs. Computing the DDLHM JSd between the first parameterization pair and every other would require 15449 computations. If the DDLHM JSd between each other pair were computed in sequence, the final parameter pair would require only one calculation; the DDLHM JSd between itself and itself, as the DDLHM JSd between the final parameter pair and every other pair would have already been computed. This results in 119343525 possible configurations to test for attenuation alone. Testing bias would require computing 42961815 permutations and multipath 75147670. These sum to 237453010 total configurations to test only each distortion type against itself, and not against each other. Given that each permutation computes in roughly a tenth of a second, a full evaluation would take three quarters of a year. Due to such a computational load exploring the full parameter space is infeasible. It is necessary to build a representative set of parameters that is a fraction of the size of the exhaustive set that can identify distortion parameters with acceptable additional error.

3.4 Properties of the Reduced Parameter Set

In reducing the state space of parameters we first chose to reduce the granularity of the incident distance values. Reducing the granularity of the incident distances may cause some loss in identification accuracy, although since the DDLHM catalogues rates of change and the lognormal function is fairly smooth, reduction in sample granularity is unlikely to result in very large estimation errors. Many localization systems work on data sampled at the meter- or foot-level. [16, 36, 20, 23, 3, 31] We have also identified that the lognormal model is extremely sensitive to distortion within the first few meters as it is losing power very quickly per unit distance. Taking this into account we propose an ensemble of at most 25 distinct range values are necessary, 1 sample per foot starting at one meter from the transmitter for the first 20 feet and then every 5 feet up to 50 feet, with one additional sample at 60 feet. This arrangement puts a premium on close samples when the signal is degrading quickly at a granularity common in the literature and less on farther samples when the signals are extremely similar. We also cap measurements at 60 feet since few indoor laterative localization systems expect to see usable signals out past 20 meters. This results in a drastic reduction in the number of configurations to test, reducing the convolution of the total parameter space from 237453010 to 1561000. Even given this reduction we found computation times to be extensive and set about to reduce the state space even further by limiting the range of distortion strength values considered.

Given our investigation of the behavior of the JSd between the DDLHM of unobstructed lognormal and lognormal with a particular distortion parameterization, we chose a series of 20 distinct distortion strength values for each type of distortion to span the parameter space as much as possible and be as diagnostic as possible. It is clear from Figures 3.4 and 3.5 that attenuation strength has the greatest effect on DDLHM JSd from the lowest parameter tested up to 2.25. We use attenuation strength parameter from 1.05 to 1.5 in 0.05 steps, and then from 1.8 to 2.25 in 0.05 steps. We leave a gap between 1.5 and 1.8 since at 1.5 the DDLHM JSd maximizes near 0.8. For the bias distortion we use -1 dBm, from -5 to -15 dBm by steps of -1, and from -17 to -24 dBm by steps of -1 as well. We leave out the values from -2 to -4 since they have a very small effect on the DDLHM JSd, as is apparent from Figures 3.9 and 3.10, spanning a 0.18 JSD range. We also leave out -16 dBm and stop at -24 dBm since -16 lies in a short JSd plateau and -24 dBm is far enough in the parameter space to see maximal JSd values. For the multipath distortion we use 7.5% multipath strength in steps of 2.5% up to 12.5% and from 30% to 62.5%. We also use 17.5, 22.5 and 27.5 precent strength multipath in 5% steps as those ranges span the worst DDLHM matching areas, resulting in all but universally higher values with little useful discriminating differences, as is evident from Figures 3.14 and 3.15. The additional pruning of distortion strengths results in a much smaller parameter space, down to 1500 possible configurations, 4% of the extensive set's original total of 37080 different configurations. This reduces the convolutions of the non-redudant parameter space down to 239400. Before embarking on a full test of this reduced set, it is necessary to first determine if the reduced set is still diagnostic after such a sharp culling of potential parameters

3.4.1 Diagnostic Capability of the Reduced Parameter Set

In order to assess the efficacy of the reduced parameter set, we will evaluate it against the exhaustive parameter set, attempting to match parameterizations from the exhaustive set against the reduced set. Due to the aforementioned extensivity of the exhaustive set, we universally randomly selected 10000 parameter configurations to match. Since the reduced parameter set covers only 4% of the exhaustive set, it is extremely likely that parameter configurations that do not match exactly will be drawn. We computed the JSd between the DDLHM of each of the 1500 lognormal parameterizations that make up the reduced set and each of the randomly-selected parameterizations from the exhaustive set. We will regard the reduced set parameterization that results in the absolute lowest JSd as the best possible match for the randomly-selected parameterization being tested. The goals of this test are threefold: determining how well the metric works with incomplete information, informing us how to interpret a large number of DDLHM JSd results and the degree of error between improper classifications and the correct ones and how often they occur.

General Distinctiveness

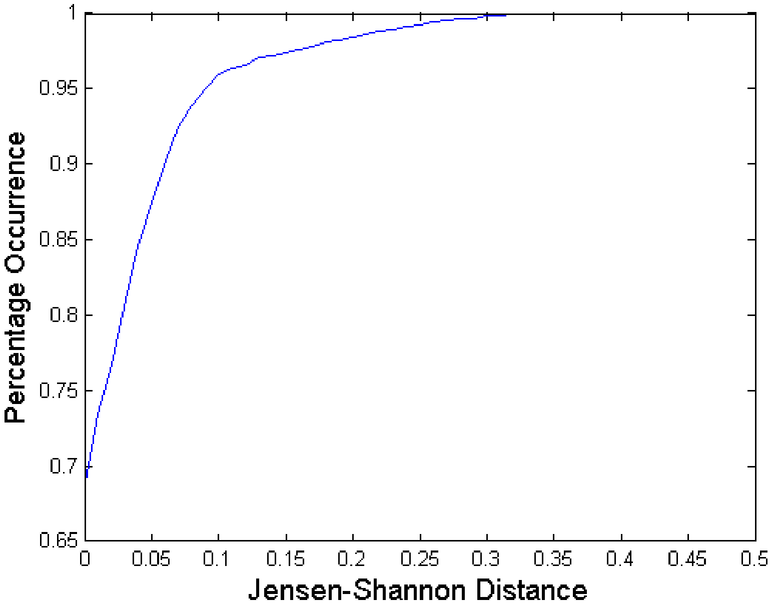


Figure 3.19: CDF on JSd between DDLHM of best match in reduced set and match made

Of the randomly-generated 10000 parameter configurations 7346, or 74%, had their distortion type correctly identified. Of those, the exact distortion strength was correctly identified in 2113 configurations and the exact incident distance in 604. Given that these calculations were done using 4% of all the available data, we find this level of matching acceptable. These results also demonstrate the capability of the DDLHM metric to correctly identify distortion types using a very small slice of available data. Beyond exact matches however, we found the DDLHM to be particularly resilient. In order to place these measurements in context we determined the configuration of the reduced parameter set that was the most like each of the randomly-generated test configurations, the best one that could have been chosen, and computed the JSd between the DDLHM of the best available match and the actual match made for all matches. Figure 3.19 depicts the CDF of these values. Even in the face of errors in distortion type, strength and incident distance, 95% of all matches made were at most 0.1 JSd away, and 85% were at most 0.05 JSd away from the best possible match afforded by the reduced set. Referring to Figure 3.2 can put these values into further context; a JSd of 0.1 is approximately the difference between two Gaussian distributions with a standard deviation of 1 and means that differ by 1. We find this level of estimation accuracy reasonable given the extreme reduction in the parameter space and consequent reduction in time to compute a match. Now that we have a reasonably diagnostic parameter set of a reasonably computable size, we must next determine the bounds of its behavior.

Parameter Distinctness per Distortion Type

While it is well enough to relate JSd measures to the Normal Gaussian for gross judgments, more textured knowledge is necessary in order to understand how the DDLHM relates to changes in the lognormal model directly. Without an understanding of the sensitivity of the JSd to changes in parameter values, it is difficult to judge what a particular JSd means. In order to determine what JSd range indicates a good match, we will match each parameter pair from the reduced set against every other parameter pair for the same distortion type and record the JSd between their DDLHMs. Since a change in either parameter can cause a change in JSd, it is necessary to analyze both in terms of the other. To do so, we will hold one parameter fixed and vary only the other, noting and dividing the absolute change in JSd by the absolute change in the free parameter, resulting in a collection of JSd per free parameter sensitivities. We will then compute the CDF on these sensitivities and systematically move to the next value for the fixed parameter. This will result in a CDF per value of the fixed parameter on the JSd per free parameter sensitivity. These CDFs together will describe the behavior of the JSd sensitivity in relation to the parameter held fixed, so that an expectation of the variability of JSd sensitivity can be determined for a given parameter and distortion.

Attenuation

In Figures 3.22 and 3.23, we held incident distance fixed and varied distortion strength only. Attenuation strength varies in JSd sensitivity from 0.22 to 1.5 JSd per attenuation multiplier on average. Since the average difference in the attenuation strength parameters is approximately 0.05, in real terms these sensitivity values mean that any change in attenuation strength will result in an average increase in JSd ranging from 0.011 to 0.075. As can be seen in 3.22, the decrease in JSd sensitivity is approximately linear in relation to incident distance from the transmitter, which stands to reason.

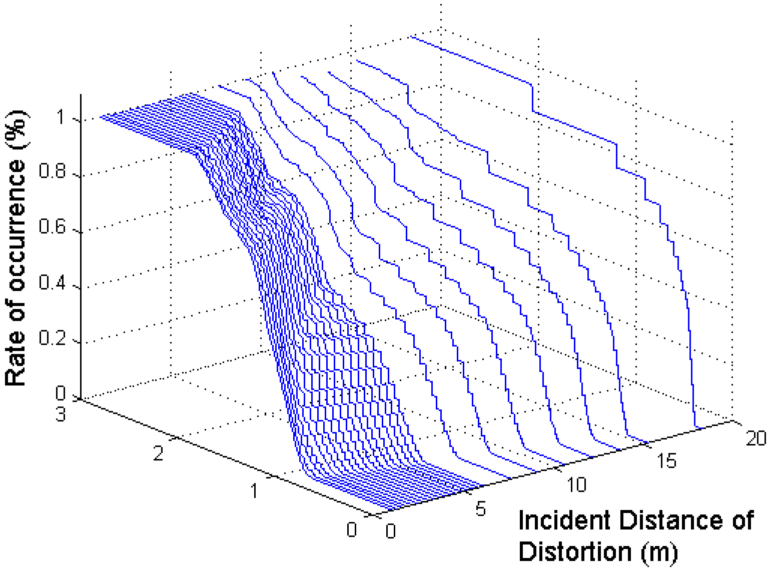


Figure 3.20: Self-Sensitivity of Attenuation Reduced Set Parameterizations: CDF on JSd per Meter of Incident Distance

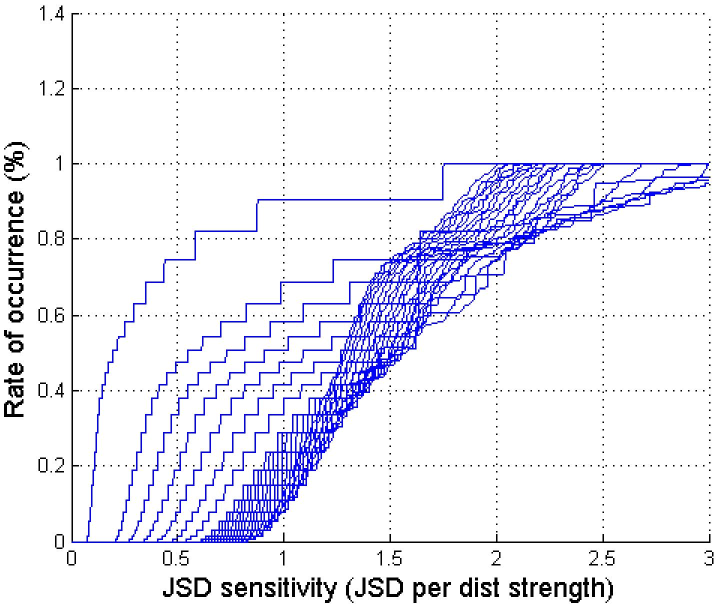


Figure 3.21: Self-Sensitivity of Attenuation Reduced Set Parameterizations: CDF on JSd per Meter of Incident Distance: alt. view

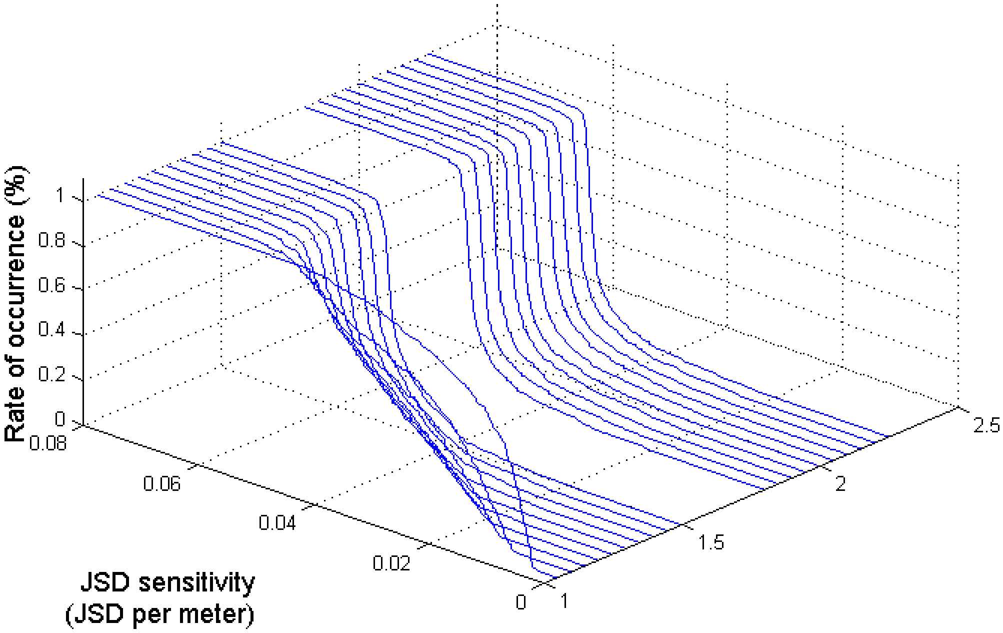


Figure 3.22: Self-Sensitivity of Attenuation Reduced Set Parameterizations: CDF on JSd per Distort Power Parameter

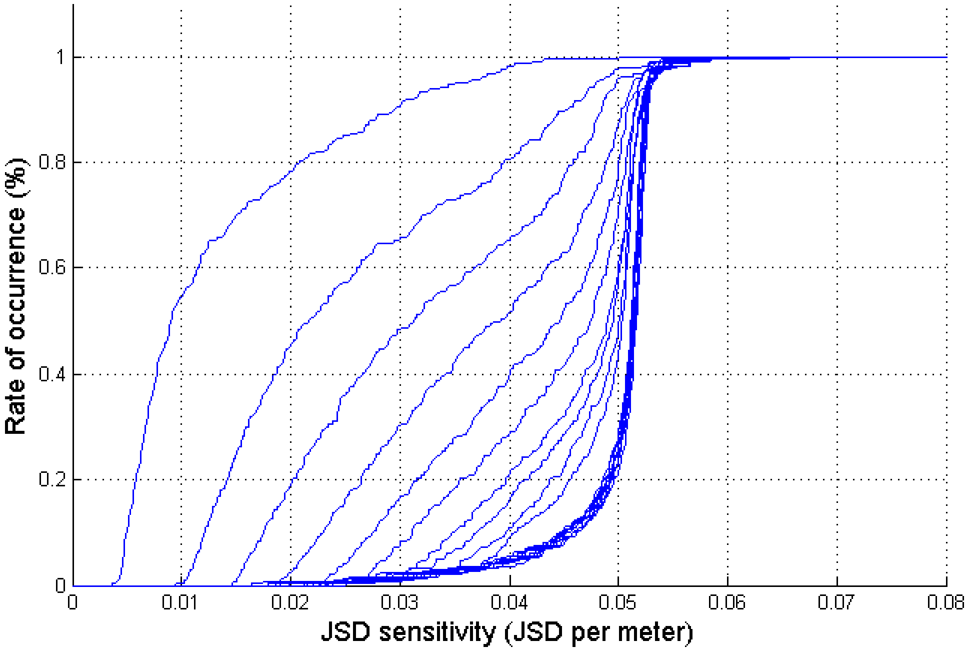


Figure 3.23: Self-Sensitivity of Attenuation Reduced Set Parameterizations: CDF on JSd per Distort Power Parameter: alt. view

Since the attenuation distortion continuously alters the lognormal model it is reasonable to presume that the longer the distortion is applied, at any strength, the larger the resulting JSd difference would be. Since 80% of all JSd per distortion strength values are below 2, any match of 0.1 JSd or lower on attenuation incident distance is likely to be a good match.

In Figures 3.21 and 3.20, we held distortion strength fixed and varied incident distance instead. Attenuation incident distance varies in JSd sensitivity from 0.01 to 0.05 JSd per meter, on average. Since the average difference in incident distance in the reduced parameter set is 0.72 and the median incident distance is 0.31, we will use 0.40 as an incident distance to compare against. Given an incident distance of approximately 0.40, we expect that any change in incident distance will result in a JSd change of 0.004 to 0.02. From Figure 3.21 it is apparent that a JSd sensitivity of 0.05 is a definite upper limit and fairly characteristic for higher attenuation strengths. From Figure 3.20, we can see that the very low sensitivities only apply to very low attenuation strengths and converge to 0.05 as the distortion strengths increase. We will then prefer the upper edge of the expected sensitivity range. From these ranges we can determine that distortion strength has a much greater effect on the JSd than incident distance when matching DDLHM profiles. An attenuation parameter match can be incorrect by up to two meters of incident distance and still fall within the criteria of a good match for distortion strength. Since incident distance is much less sensitive the attenuation strength, we will maintain that a match of 0.1 JSd or less is a good match, based on the behavior of attenuation distortions, while a match of 0.2 JSd or less is reasonable, based on the general behavior of the JSd metric.

Bias

In Figures 3.26 and 3.27, we held incident distance fixed and varied distortion strength only for bias distortions. Bias strength varies in JSd sensitivity from 0.25 to 0.37 but for a few outliers that occur at extreme distances only. The bias distortion strengths in the reduced set almost all differ by 1, so in this case the JSd sensitivity per distortion strength is itself the resulting JSd change. Given that 0.5 JSd results from comparing a distribution to a null distribution, and that 80% of all JSd sensitivities are below 0.43, we can presume bias distortions are extremely sensitive to the distortion strength, at nearly any incident distance. Any bias match that is in any way reasonable should have a very low JSd since we can expect a very large difference whenever the strength value is changed. Due to the sensitivity of bias to distortion strength, we will regard any match with less than 0.25 JSd as exact on the distortion parameter.

In Figures 3.25 and 3.24, we held distortion strength fixed and varied incident distance instead. Very much like attenuation, bias JSd sensitivities per meter over all distortion strengths range

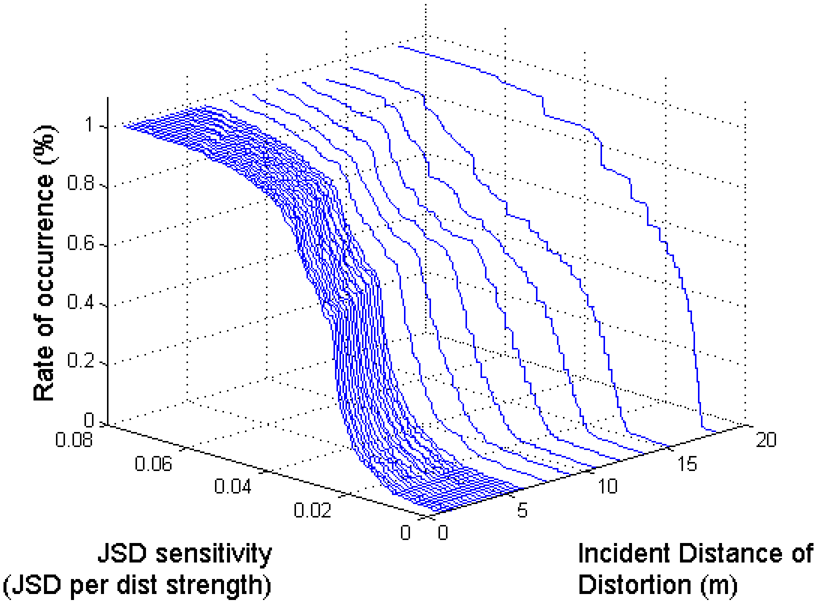


Figure 3.24: Self-Sensitivity of Bias Reduced Set Parameterizations: CDF on JSd per Meter of Incident Distance

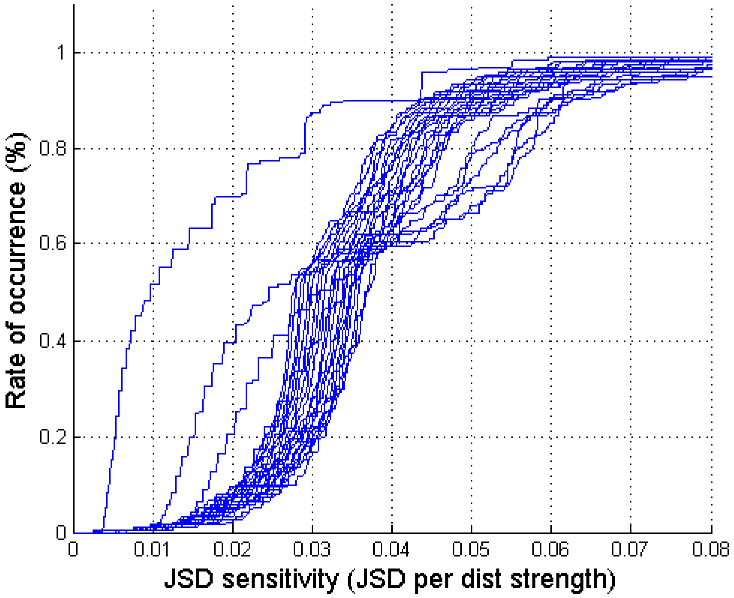


Figure 3.25: Self-Sensitivity of Bias Reduced Set Parameterizations: CDF on JSd per Meter of Incident Distance: alt. view

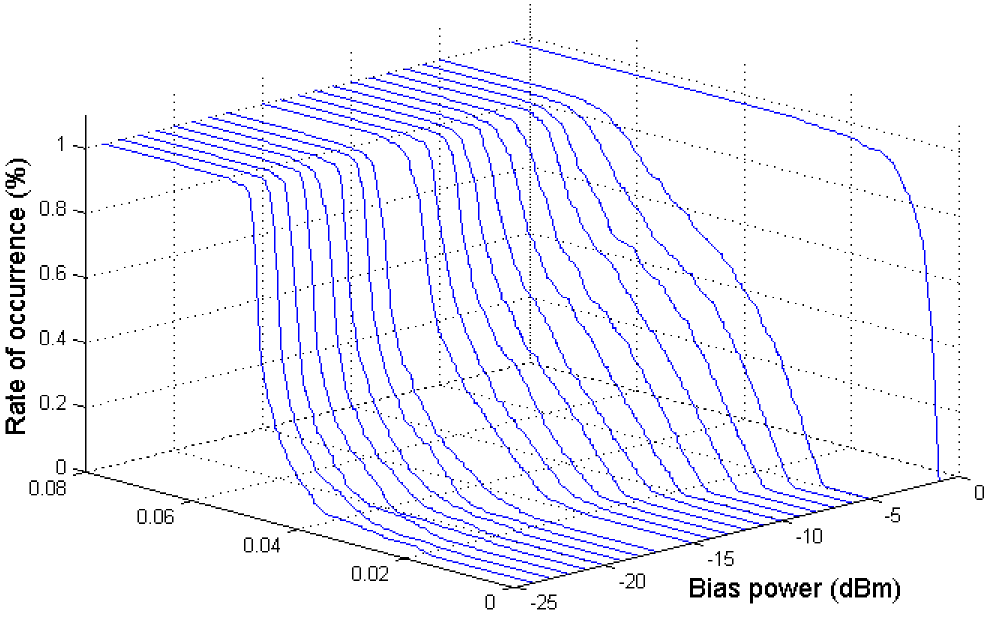


Figure 3.26: Self-Sensitivity of Bias Reduced Set Parameterizations: CDF on JSd per Distort Power Parameter

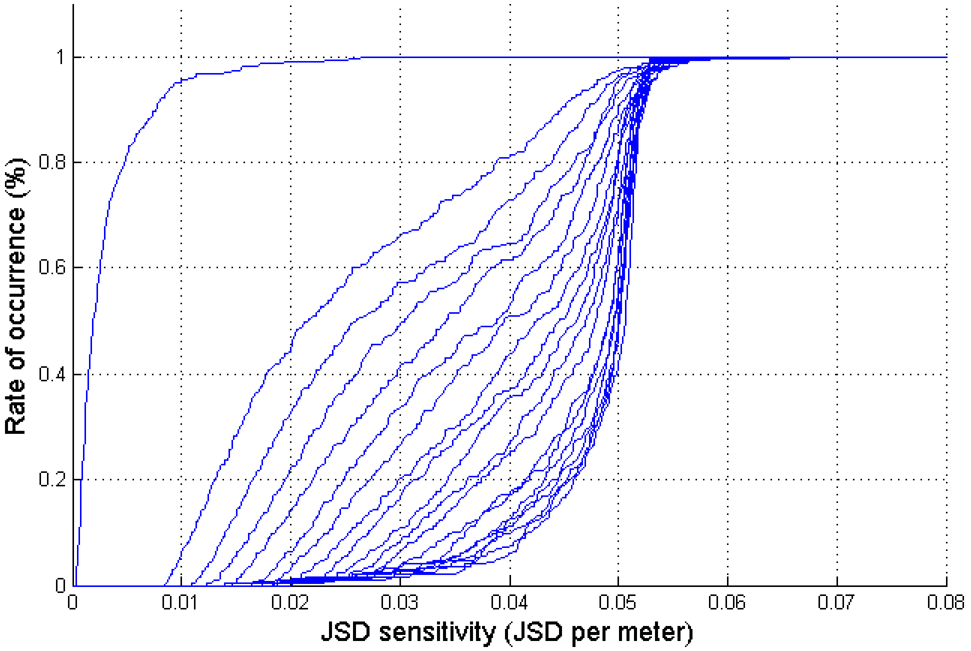


Figure 3.27: Self-Sensitivity of Bias Reduced Set Parameterizations: CDF on JSd per Distort Power Parameter: alt. view

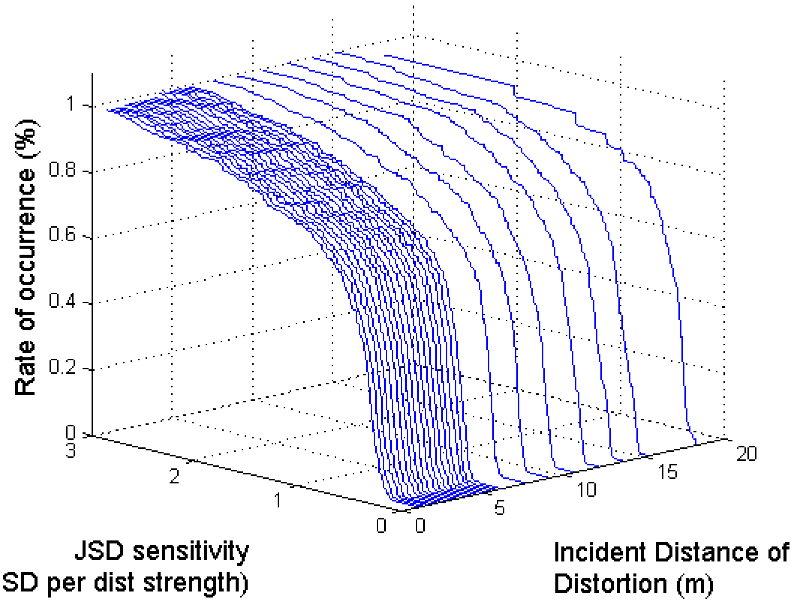


Figure 3.28: Self-Sensitivity of Multipath Reduced Set Parameterizations: CDF on JSd per Meter of Incident Distance

from 0.02 to 0.05, but for one outlier. Unlike attenuation, bias sensitivities per meter behave more consistently, bringing the median and mean performance closer together, narrowing the 80% range of sensitivities to span between 0.04 and 0.05 for bias. Using our incident distance granularity of 0.40, we can expect a JSd increase of between 0.016 and 0.02 for each 0.4 meters of incident distance beyond an exact match. Due to the extreme difference between distortion strength and incident distance sensitivities, it is quite likely that all of a JSd on a bias match below 0.25 JSd is attributable to incident distance error entirely. Since bias strength is so much more sensitive and descriptive than incident distance, we will regard any bias match with a JSd below 0.25 JSd as a good match.

Multipath

In Figures 3.30 and 3.31, we held incident distance fixed and varied distortion strength only for multipath distortions. Multipath Strength varies in JSd sensitivity from 0.25 to 0.56 on average, although the tail of the CDF widens considerably with the 80th percentile spanning a range from approximately 0.5 up to 1.5, so we will use 1.2 as an approximate expected JSd sensitivity bound. The multipath distortion strengths in the reduced set differ by 0.025 except for three that differ by 0.05, so we will use 0.027 as an approximate expected granularity of strength parameter difference. These sensitivity parameters would result in an expected increase in JSd of 0.032 per parameter

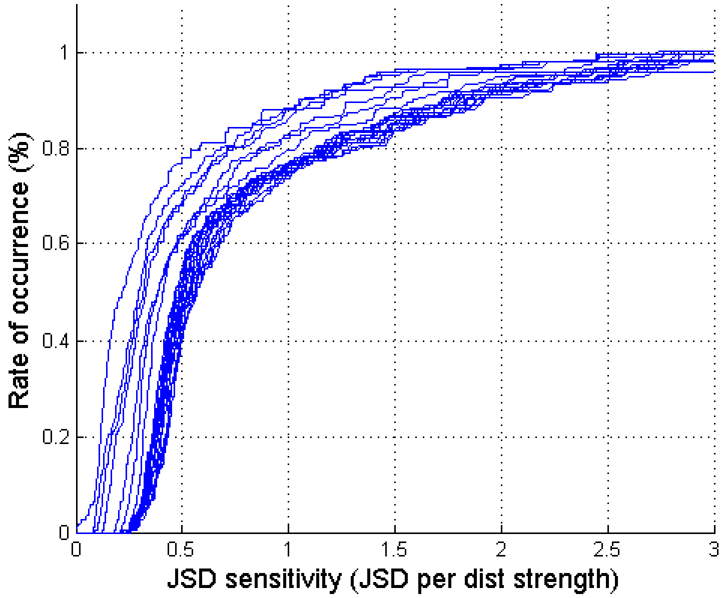


Figure 3.29: Self-Sensitivity of Multipath Reduced Set Parameterizations: CDF on JSd per Meter of Incident Distance: alt. view

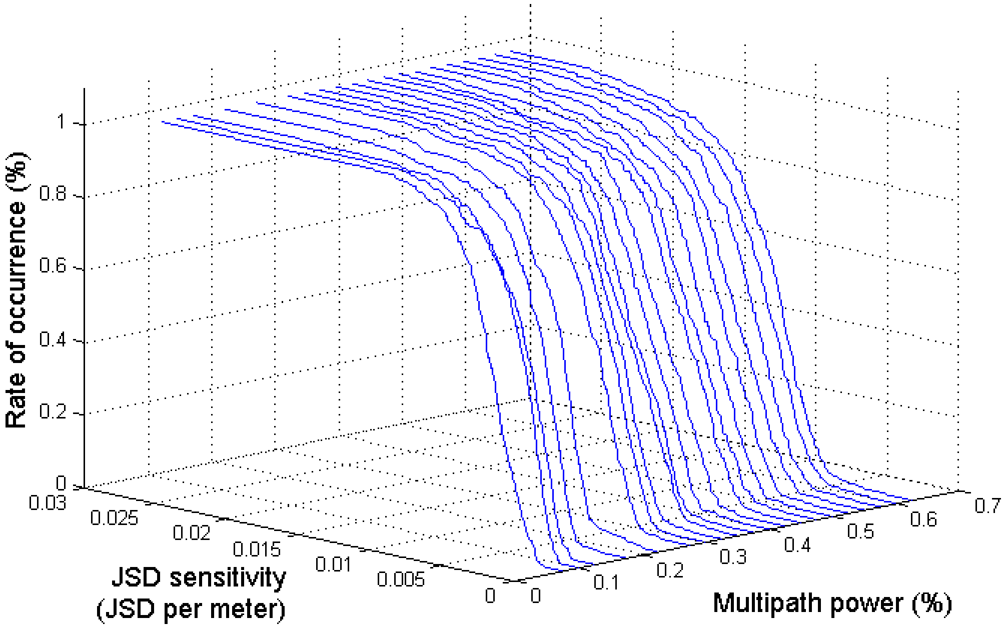


Figure 3.30: Self-Sensitivity of Multipath Reduced Set Parameterizations: CDF on JSd per Distort Power Parameter

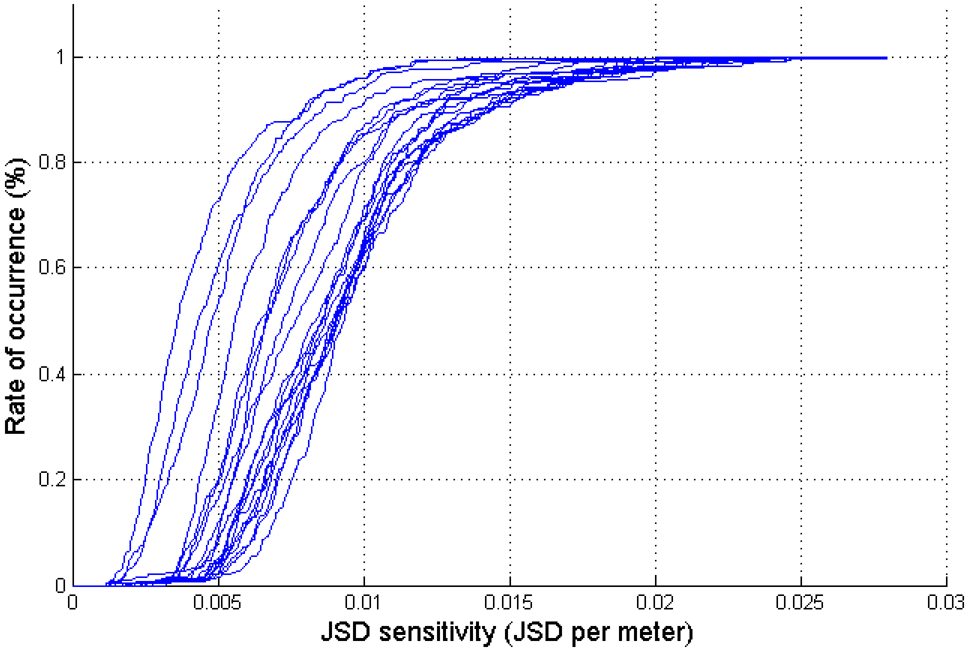


Figure 3.31: Self-Sensitivity of Multipath Reduced Set Parameterizations: CDF on JSd per Distort Power Parameter: alt. view

difference. While the parameter sensitivity is fairly high, the absolute difference of each parameter is so low that multipath distortions seem to have relatively little sensitivity to the strength parameter no matter which incident distance parameter is chosen. Multipath incident distance varies in JSd sensitivity between some extremely low values, with 80% of all values in Figure 3.29 falling below a sensitivity of 0.0125 JSd per multipath strength parameter, resulting in an expectation bound of 0.005 JSd per incorrectly-matched incident distance quanta. Both multipath parameters seem to be fairly insensitive to overall match performance.

Multipath proves to be particularly recalcitrant to analyze in such a manner, although it is not unexpected. As can be seen in Figures 3.17 and 3.18, there are only a few very distinct configurations of multipath parameters that result in very large, easily-detectable departures from unobstructed lognormal. Since most multipath distortion parameter sets result in propagation patterns very similar to unobstructed lognormal, it stands to reason that most multipath distortion parameter sets would result in propagation patterns very similar to each other. Hence, there is very little informational difference between the DDLHMs that result from most multipath parameter sets, and the total sensitivity seems quite low even though some particular configurations result in very strong deviations from unobstructed lognormal. We will then regard a JSd of 0.25 or better a reasonable match to a multipath configuration, although a very low JSd of 0.05 or less would be necessary to indicate a good parameter match.

Overall

From the above we can conclude that distortion strength is a much more descriptive parameter than incident distance for all distortion types considered. Bias parameterizations are the most distinctly identifiable, with the highest sensitivities for both distortion strength and incident distance amoung all distortion types, and a good parameter match indicated by a JSd of 0.25 or less. Attenuation is less exactly distinguishable, with a good parameter match indicated by a JSd of 0.1 or less, although attenuation configurations are much more distinct than multipath. Due to the strikingly marginal JSd differences between multipath and unobstructed lognormal DDLHMs, multipath has a particularly pacific sensitivity range. Since nearly any multipath parameter configuration is fairly difficult to tell apart from most others, a very low JSd of 0.05 or lower is necessary to indicate a solid match. We find these values compelling since when testing the matching capability of the reduced versus the extensive parameter set in subsection 3.4.1 we found 85% of all matches were within 0.05 JSd of the best possible match. Since this value bound is well below or within the necessary matching criteria for attenuation, bias and multipath parameter configurations, we presume that most parameter matches will be fairly unambiguous on physical data once the dominant distortion type has been identified.

While we have established the JSd bounds to ensure identifiablility and distinctness between different parameterizations of the same distortion type, we have not determined how different we expect the distortion types to be from each other. It could be the case that some range of parameterizations of one distortion type are very similar to a range of another. Since the dominant distortion type is not known a priori when analyzing physical data sampled from a live environment, it is necessary to identify it first, in which case we must know how similar all parameterizations of all distortion types are to each other in order to inform a confidence judgment based on JSd values between the DDLHM of the sampled data set and all distortion parameterizations of the reduced set.

Parameter Distinctness across Distortion Type

Due to the much stronger JSd sensitivity of distortion strength determined above in subsection 3.4.1, and we want to determine how similar two different distortions' parameterizations can be, we will compute the JSd between all distinct distortion strength pairs for two different distortions at the same incident distance. In order to determine how distinct each pair of distortion strengths are across all incident distances, we will compute the entropy of the discrete distribution of all JSd values computed per distinct distortion strength pair. If two different distortions at some strength have extremely similar JSd values between their DDLHMs across all incident distances, the entropy of the distribution of the JSd values will be very low, indicating that it would be difficult to tell the distortions apart by JSd alone in live data. If they have very different JSd values instead, the entropy will be much higher, and it would be quite likely that they would be very distinct and fairly easy to discriminate between in live data.

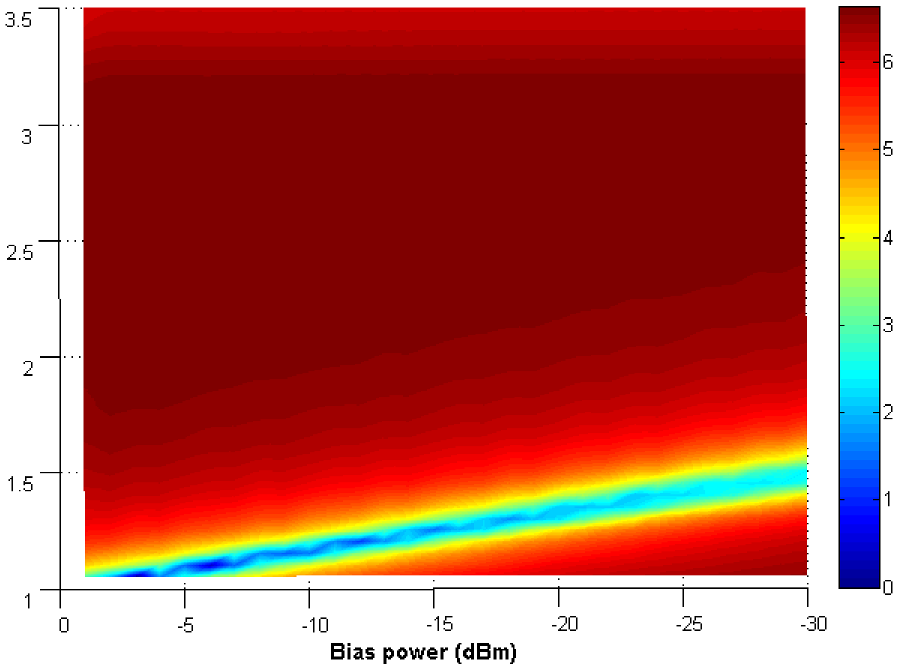


Figure 3.32: Distinguishability of Distortion Types: Entropy between Bias and Attenuation

Overall, most configurations of the reduced parameter set had very high entropies across distortion type. As can be seen in Figures 3.34 and 3.35, only a thin slice consisting of the lowest attenuation strength parameters indicates JSd performance anywhere similar to multipath distortions, with only a very small range between 10 and 15% multipath strength and 1.05 to 1.15 attenuation strength has an entropy of approaching 2.5. This is due to the fact that very low multipath strength results in several shallow dips below unobstructed lognormal rather than deep fades. The sum total of these small dips is approximated by the effect of very low additional attenuation. Beyond these very low values, the two distortion types quickly assert their distinctive behaviors. making attenuation and multipath distortions fairly easy to differentiate between. Multipath and bias distortions have a wider similarity range, from Figures 3.36 and 3.37, we can see that bias distortions below -4 dBm and multipath strengths below 1.5 have entropies that range between 2.1 and 2.5. This is due

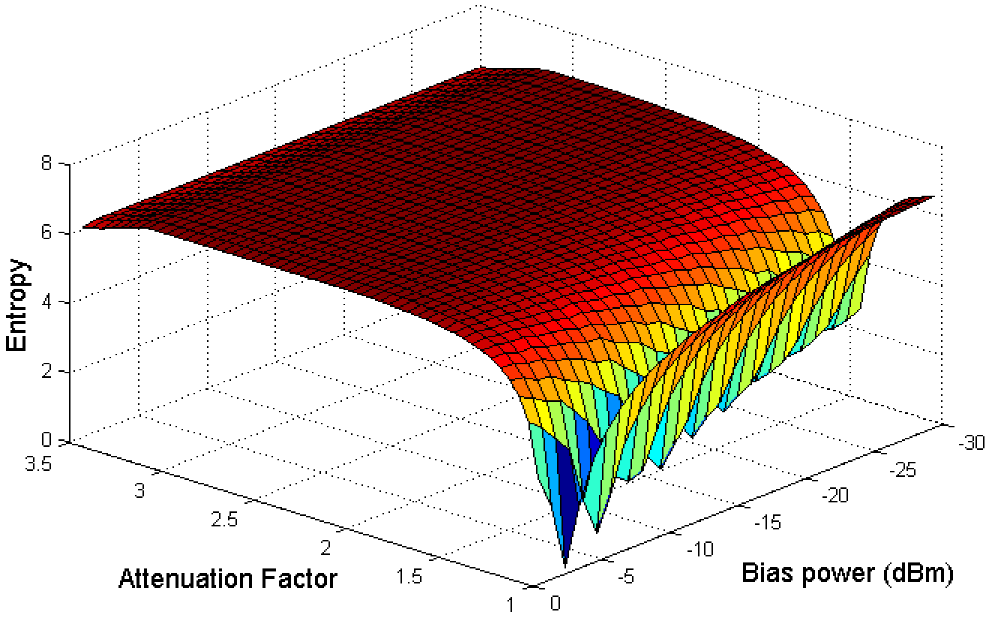


Figure 3.33: Distinguishability of Distortion Types: Entropy between Bias and Attenuation: alt. view

to the fact that a small, immediate loss in signal power caused by a bias distortion has a similar effect on the DDLHM as the sum total of a few shallow multipath fades. Beyond this single region, both distortion types decorrelate quickly, becoming much more distinct.

Attenuation and bias distortion types have a much more interesting interaction. From Figures 3.32 and 3.33, we can see a strip running across all bias strength values occasionally reaching entropies of almost 0, indicating the two distortions have nearly identical behavior at those strengths over all incident distances. This does not mean that the JSd between the DDLHMs of the propagation patterns generated by the chosen parameters are low, only that they are nearly identical. Since these configurations have nearly identical performance, it would be extremely difficult to distinguish between them in live data. Additionally, of the 26% of distortion types misidentified in 3.4.1, over 82% came from this range of bias and attenuation parameters. Since a bias distortion is a one-time drop in signal power at a given incident distance, and an attenuation distortion begins applying a depressed version of unobstructed lognormal propagation at a given incident distance, the immediate drop in signal power caused by encountering an attenuating object can seem very much like a single drop due to bias, especially if the rest of the attenuative effect is fairly mild.

Except for very low strengths of the distortions and several pairs along the similarity line between attenuation and bias, all JSd distributions had an entropy above 2.5 along their least sensitive

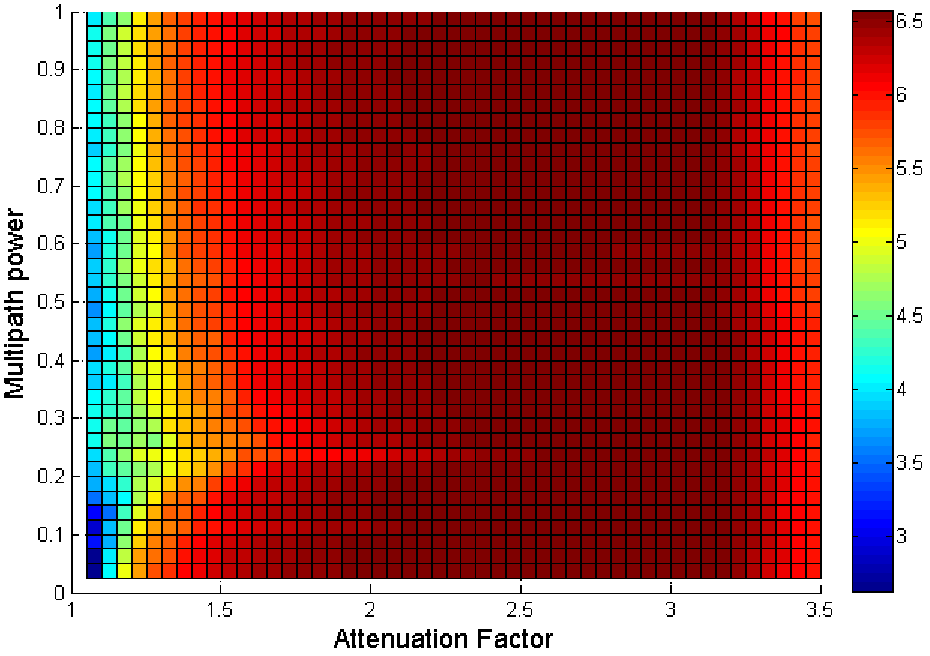


Figure 3.34: Distinguishability of Distortion Types: Entropy between Multipath and Attenuation

dimension, incident distance. Demonstrating that, for the vast majority of cases, that different distortion types of the reduced parameter sets are quite distinctive and result in very different patterns of JSd values. Given that this is the case, the strongest distortion applied to live data should be quite eminently identifiable by comparing its DDLHM to the entire reduced parameter set and selecting the distortion and parameter configuration with the minimal JSd.

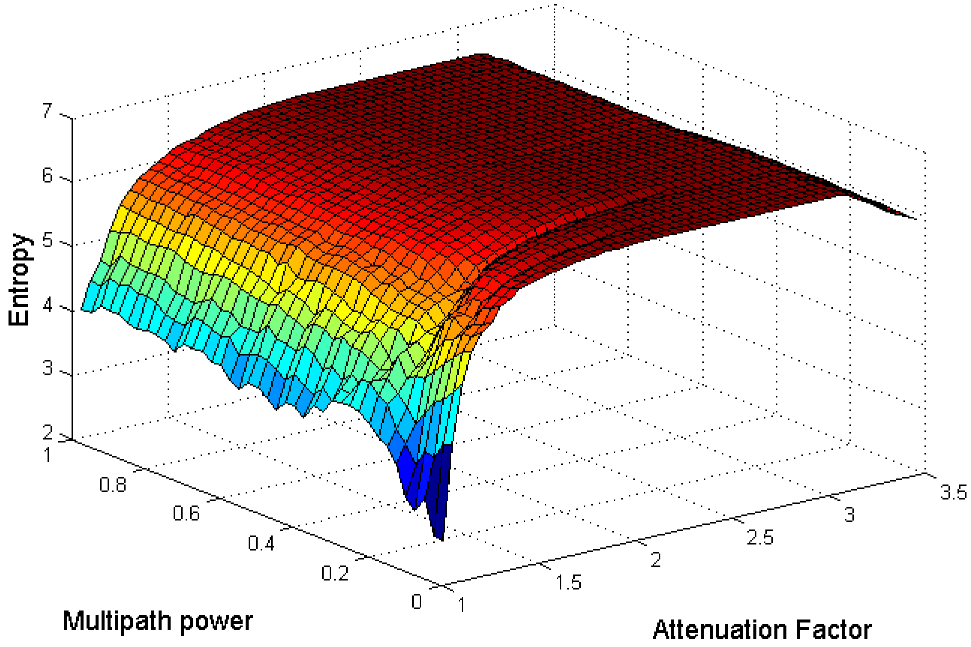


Figure 3.35: Distinguishability of Distortion Types: Entropy between Multipath and Attenuation: alt. view

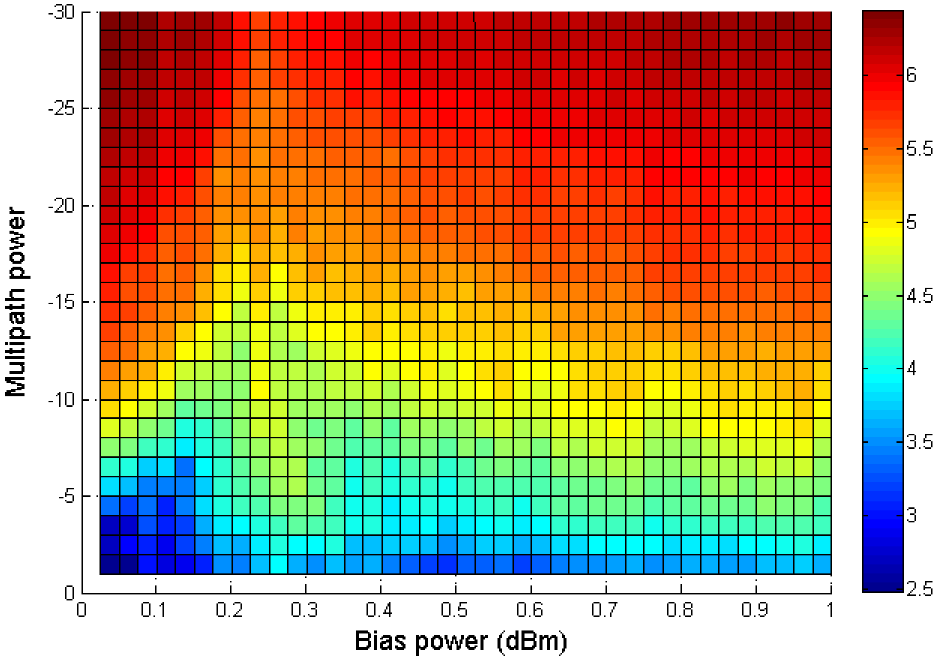


Figure 3.36: Distinguishability of Distortion Types: Entropy between Multipath and Bias

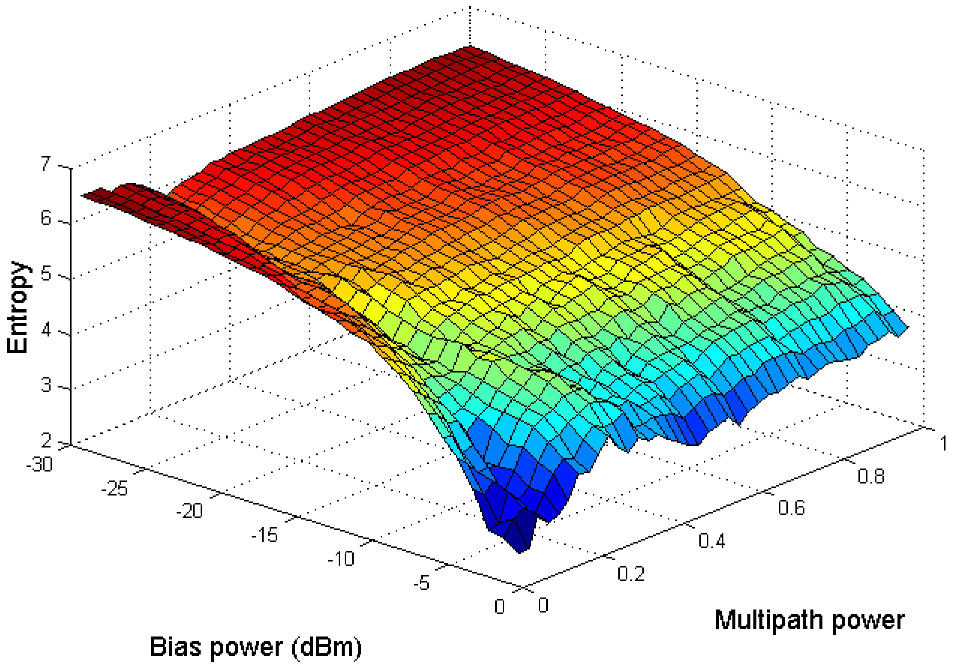


Figure 3.37: Distinguishability of Distortion Types: Entropy between Multipath and Bias: alt. view

Chapter 4
Localization Algorithm Analysis

There is no standard method to gauge laterative localization performance that is environmentally and algorithmically agnostic. The common method to determine the capabilities of a localization algorithm is to test it on live data. This indelibly ties the error results to a given algorithm and environment. This makes it impossible to compare two algorithms without testing them in the same environment. Since the data captured is a momentary snapshot of what the radio environment was at the time, any results computed are momentary and may not be characteristic of the environment. Even so, it would be unclear if a given algorithm's performance is due to some peculiarity of the data or of the algorithm itself. This lack of a diagnostic metric forces any changes to a localization algorithm to be either heuristic or dependent only on general statistical arguments. Since a modification of an algorithm can itself be viewed as a separate algorithm, and by our reasoning above any localization result is singular, it is not possible to determine why a given modification or algorithm produced a particular result, only that it does. The reverse is also true. While localization results can be tabulated, it is not clear if the algorithm or the environment caused the result. Such a situation makes any type of reasoned and duplicatable improvement impossible.

All laterative localization error can be cast as ranging error since, if there were no error in the ranging mechanism, a localization algorithm should compute each location exactly. By generating synthetic data with known amounts of error and systematically altering the amount and type of radio distortion applied through a representative set of error configurations, the sensitivity and error trends of an algorithm in relation to precise characterizations of ranging error can be gauged. Such an analysis would provide an evaluative benchmark to determine an algorithm's expected behavior when confronted with different types and amounts of ranging error. While this information would be useful to inform the improvement of a particular algorithm, such measurements lack context.

Laterative localization algorithms are often built on statistical or machine learning principles, each reducing error in its own way. Due to this singular nature of localization algorithms, sensitivity to ranging error is unlikely to indicate the same degree of location accuracy between algorithms. To establish a common context to determine how well a given algorithm localizes in the face of a precise amount of ranging error it is necessary to compare it against an algorithm with not only a closed-form error function, but with no inherent error reduction, so that each and every quanta of ranging error directly contributes a predictable and exact amount of localization error. An algorithm's location accuracy per ranging error parameter can then be compared to the closed-form algorithm, determining precisely how much better it performs. Two algorithms can be contrasted by comparing their relative efficacy ratios vs the closed-form algorithm.

4.1 Benchmarking Localization Algorithms

The statistical nature of most localization algorithms' laterative engines render them largely incapable of being described in closed form or even bounded effectively since the a-priori values of the distributions they estimate are unknowable. One such common rationale is to cast RSS as conditionally dependent on the distance a signal has traveled and then apply Bayes' Rule to reverse the causal direction. While this results in a convenient formulation with much prior work that can be applied, computing an error model for such a process would require the distribution on all possible transmitter locations and RSS values to be known for all environments a-priori. Localization can easily be cast as an instance of any of a number of other common classification or matching problems much to the same effect; they enable straightforward computation of results, but cause the relation between the results and data to be uninspectable. This relegates the improvement of laterative localization algorithms to general improvements on statistical algorithms.

Applying common statistical and machine learning techniques without investigating the unique cast of the laterative localization problem can lead to close modeling and analysis of data in a manner that does not reflect the conditional relation between the data and localization error. While general conclusions can be drawn about the error trends when using data collected from a live environment, the inherent opacity of the signal data makes it impossible to determine the algorithmic significance of the results. Localization algorithms depend on two major assumptions: that signal data sampled per point is diagnostic of that point, and that the ranging process, built on the lognormal model, is correct. While it is a simple matter to collect more data at a point and compute the average of data, refining the expected value, determining the fitness of the ranging process and the suitability of the lognormal model is less apparent.

Since we can not know the exact reason for each signal value and, consequently, each localization result, a laterative localization algorithm can only be rationally examined by exercising it on structured data. Using the parameters determined during the construction of the DDLHM evaluation process above, it is possible to engineer a test process using a suite of propositional sample vectors with specific error characteristics in order to inform a direct conditional relation between aspects of signal sample data and trends in algorithmic accuracy.

4.1.1 Parameters of the Benchmark Environment

As noted above, we can not know why an algorithm produced a given result, only that it did. The difficulty comes in not knowing the exact environmental properties that cause any given signal. In order to have an environment whose parameters are directly and exactly decidable, we constructed a synthetic one. Due to the size of most indoor environments, we limited the size of our synthetic environment to a 20 by 20 meter square, with the LMs located outside the sample area at rows 0 and 21. Since most localization algorithms' data is sampled on a granularity of feet to meters, we generated samples per meter. We arranged the first LM bisecting the square horizontally and the other two at the extreme corners opposite the central LM, forming an equilateral triangle. This organization is known to be quite stable and error-reducing for localization algorithms using three LMs. [13] We then computed the RSSI for each LM, and fed that data as testing data to be located, inducing errors only the bisecting LM's testing data. We then used the parameters decided upon for the reduced parameter set above as a basis to generate testing data with a known amount of each identified major distortion type. For algorithms that require training data, we calculated the RSSI for the entire environment with no error except for the column of coordinates the LM bisecting the space lies on. For all these coordinates, we generated training data with distortions applied to coordinates appropriate to the parameters and will tested the algorithm using a leave-one-out method, so that the algorithm can make use of distorted training data to better classify and reduce consequent error. Any resulting localization is then solely an aspect of only the localization algorithm.

4.1.2 Algorithms Evaluated

We will evaluate a suite of algorithms characteristic of model-based laterative localization and localization in general. We will test both pointwise and laterative algorithms. Pointwise algorithms directly relate the signal values recorded during training to the coordinates where they were sampled. While they do not incur the model dependence of laterative algorithms directly, their error model is just as impenetrable. RADAR, Simple Point Matching (SPM) and Area Based Positioning (ABP) are all pointwise algorithms. The laterative algorithms will we consider are M1 and M2. Laterative algorithms include at their core a lognormal model that is parameterized by some process to convert sampled training and testing data into ranges during their ranging phase. These ranges are then used to compute transmitter coordinates during the lateration phase. In some cases, the ranging and lateration phases are computed at the same time.

Pointwise Algrithms

The RADAR Algorithm

The RADAR algorithm is a pointwise algorithm. [9] Training data is collected at known coordinates. Testing data is compared numerically to the training data and the training data fingerprint that has the smallest sum total signal difference is determine to be a match. The coordinates where the matching fingerprint was sampled are reported as the coordinates of the transmitter. We include the RADAR algorithm as a control since it makes no attempt whatsoever to model the relation between sampled data and location.

The SPM Algorithm

The SPM algorithm is an extension on the RADAR algorithm. [16] It functions in a similar manner, but addressed a particular deficiency or RADAR, namely that it can not localize a signal to any coordinates that were not included in the training data. This strongly limits the potential accuracy of RADAR. SPM addresses this issue by first imposing a grid on the environment and populating unsampled grid tiles with data based on the interpolation of data in the sampled tiles. It then matches testing data against all tiles, interpolated or sampled, and returns the center of the tile with the most similar signal fingerprint as the transmitter's location. While SPM computes in a pointwise method, the addition of interpolation makes it at least partially model based. The SPM algorithm makes the presumption that Delaunay Interpolation correctly models the propagation of 802.11 between tiles in the environment.

The ABP Algorithm

The ABP algorithm is similar to SPM. [16] It uses the same environmental gridding and interpolation, however it computes location differently. It makes a further presumption that error is due mainly to environmental noise, and adjusts its reckoning process accordingly. Instead of matching tiles by value directly, ABP regards the training data per-grid as the mean of a Gaussian distribution. It then computes the likelihood that the testing data came from that tile using a preset standard deviation, per LM. The top-k tiles per LM are then matched across LMs. The probabilities across the selected group of tiles is renormalized, the tiles are sorted by their distance from a preset confidence bound, and center of the best-matching tile is returned as the transmitter location.

Laterative Algorithms

The M1 and M2 Algorithms

The M1 algorithm is a laterative algorithm. [16] It seeks to use Bayes' Rule to reverse the conditional dependence and cast signal as causing distance. The difficulty it address is that in order to calculate in such a direction, the likelihood of the testing data being transmitted from any coordinate in the environment must be known. Since any location is equally probable across the plurality of all environments, this likelihood is described by a universal distribution, making the final equation incalculable. The M1 algorithm then estimates probabilities proportional to the actual ones by using a modification of slice sampling in order to determine the most likely distance to cause a given signal. The parameterization to the lognormal path model that would result in such a distance is then recorded, and the parameters that would result in the minimal error across all training data recorded are estimated and applied to the testing data to determine range. The M2 algorithm is a modification of the M1 algorithm that uses unsupervised learning. Rather than tagging each signal sampled from the environment with the location where it was recorded, the M2 algorithm draws propositional values for each parameter from a distribution, all that is required are enough distinct signal samples from the environment. The rationale being that, if the lognormal model does indeed describe the relation between signal and distance, given enough random parameterizations the correct set should be drawn. Since there are any number of parameterizations that could describe a single sample, the parameters that best describe a plurality of samples should be close to correct. After calculating the most apparent set of parameters, the algorithm can laterate on any of the data.

4.2 Algorithm Error Characteristics

Having carried out the indicated computations for all algorithms described over the parameter set enumerated, we recorded the resulting meters of localization error. We organized and analyzed the results in a distortion-major manner. All results are presented in three dimensions, with height always depicting each of the 20 distinct points along the bisecting central column of points. Point 1 is one meter away from the central LM and point 20 is 20 meters away. Localization error is indicated by both the size and color of each point, and is always measured in meters. Distortion strength and incident distance are always the bottom two axes, although per graph these rotate in order to display more faces of the graph.

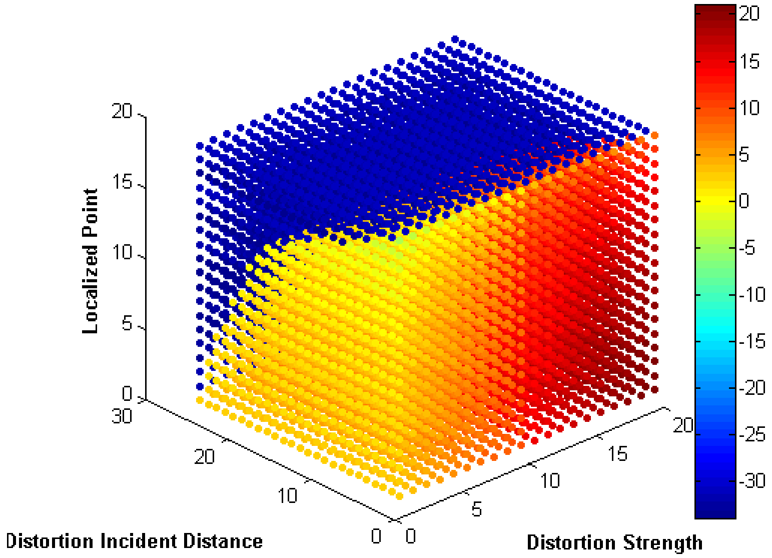


Figure 4.1: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: Rotated Trilateration

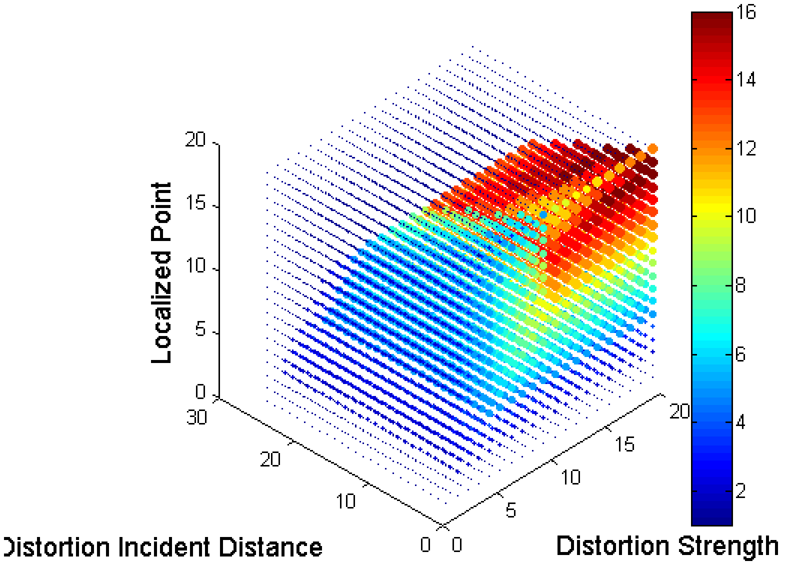


Figure 4.2: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: RADAR

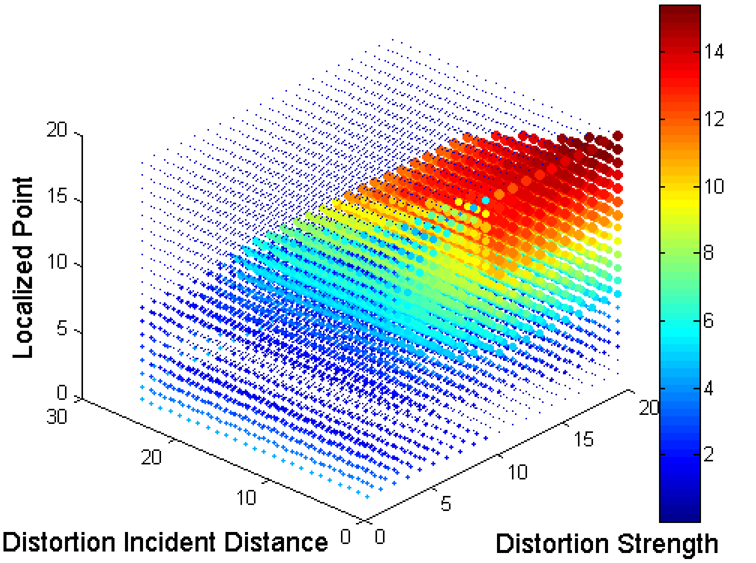


Figure 4.3: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: Area Based Positioning

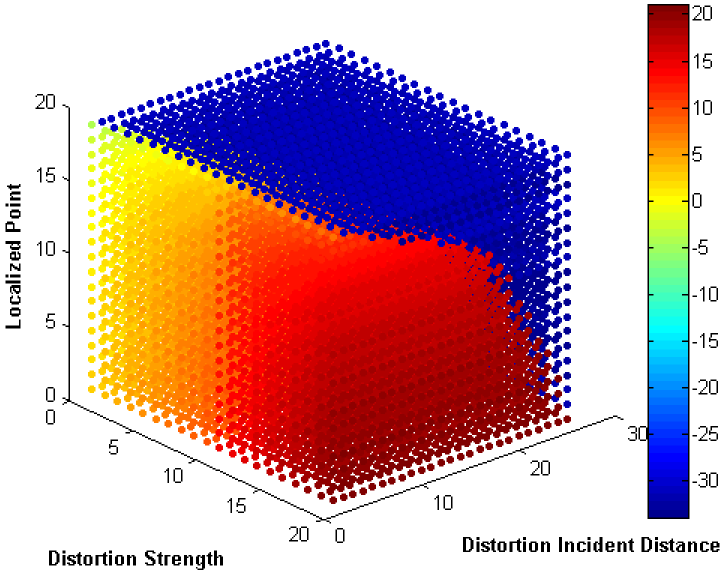


Figure 4.4: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: Rotated Trilateration: alternate view

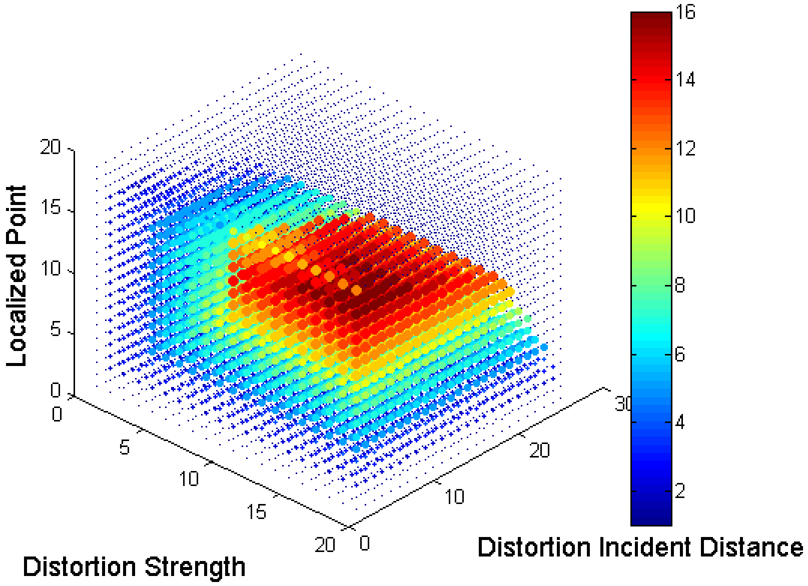


Figure 4.5: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: RADAR: alternate view

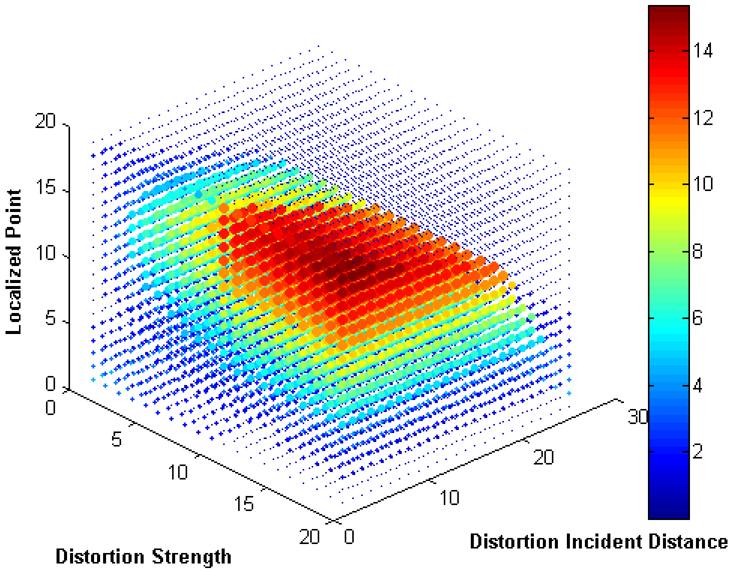


Figure 4.6: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: Area Based Positioning: alternate view

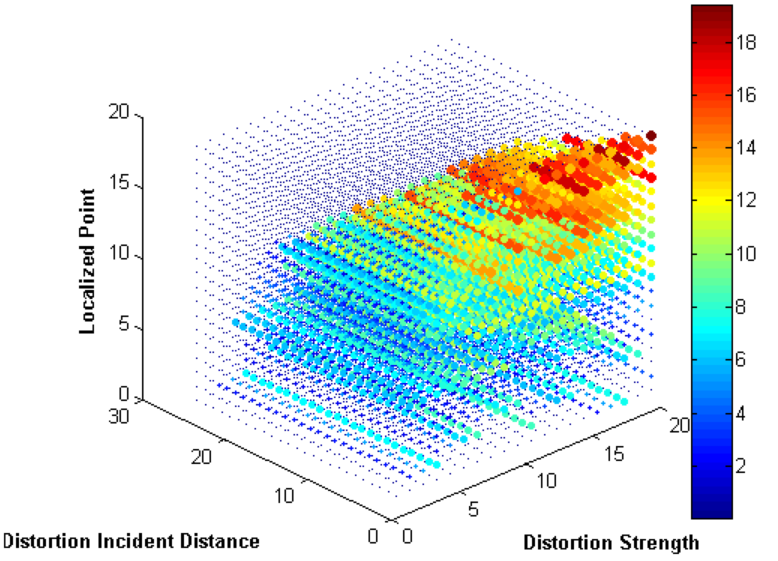


Figure 4.7: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: Simple Point Matching

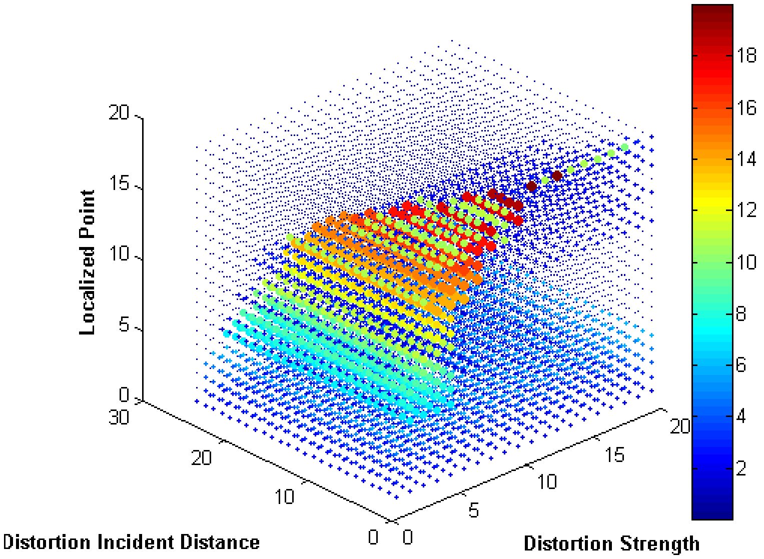


Figure 4.8: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: M1

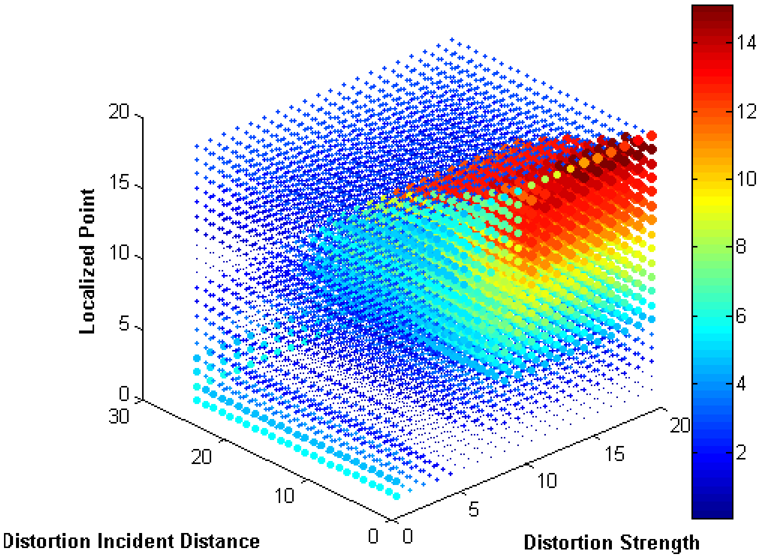


Figure 4.9: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: M2

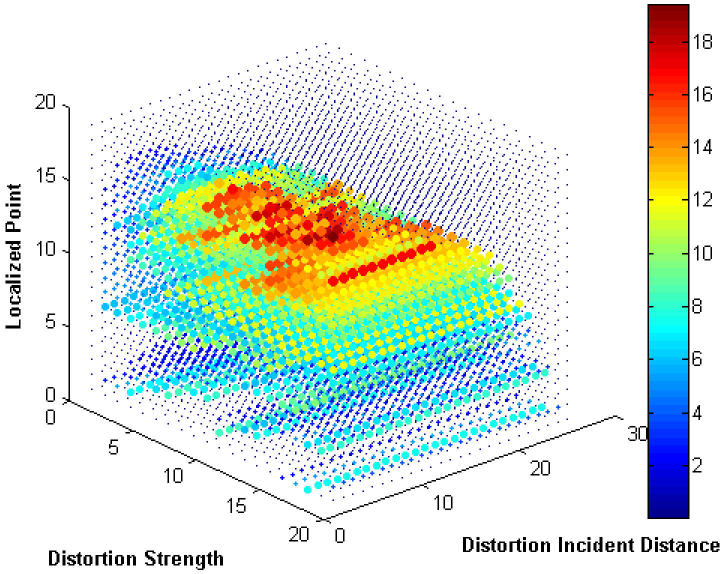


Figure 4.10: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: Simple Point Matching: alternate view

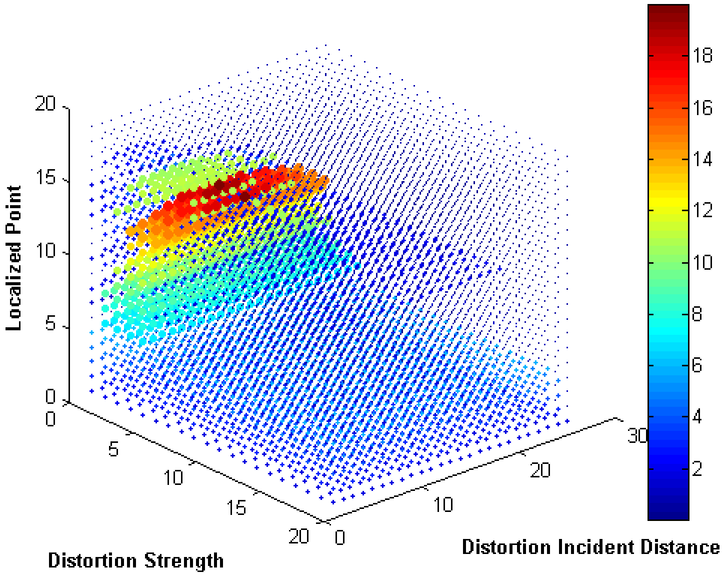


Figure 4.11: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: M1: alternate view

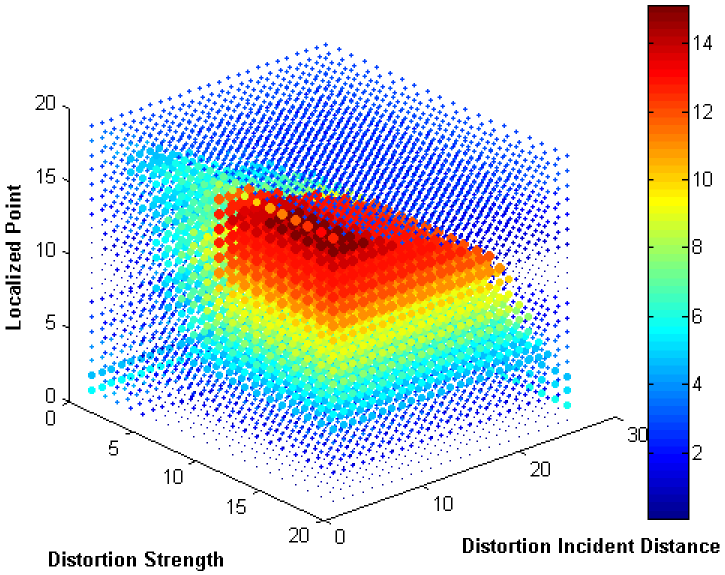


Figure 4.12: Localization Error per Descriptive Set Parameter per Point for Attenuation Distortions: M2: alternate view

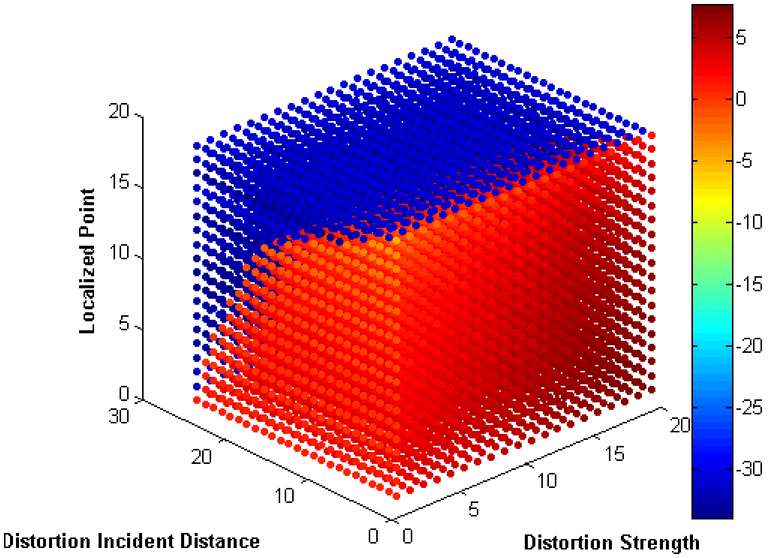


Figure 4.13: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: Rotated Trilateration

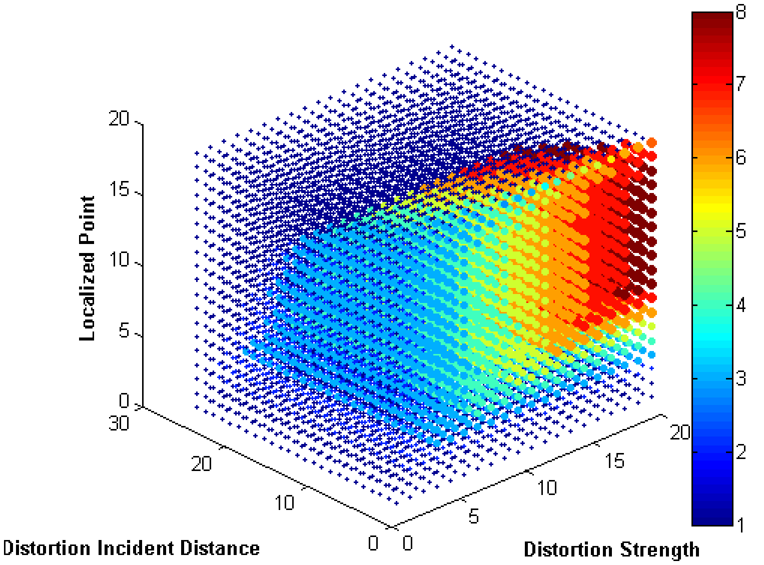


Figure 4.14: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: RADAR

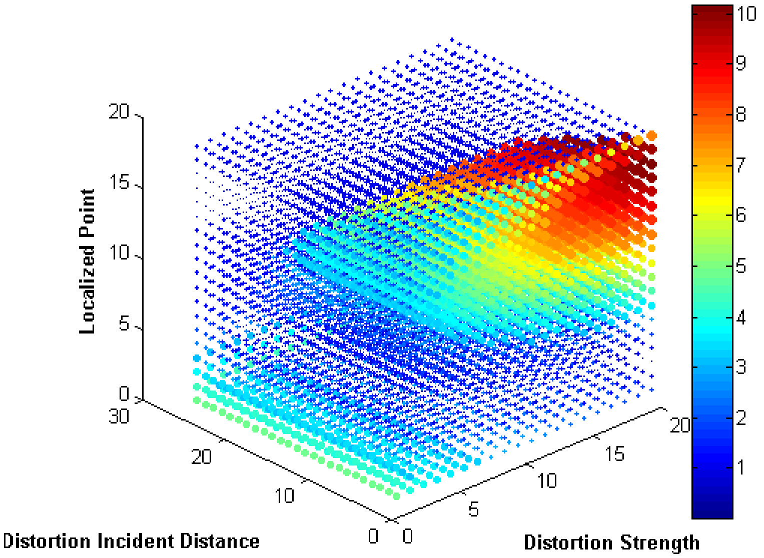


Figure 4.15: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: Area Based Positioning

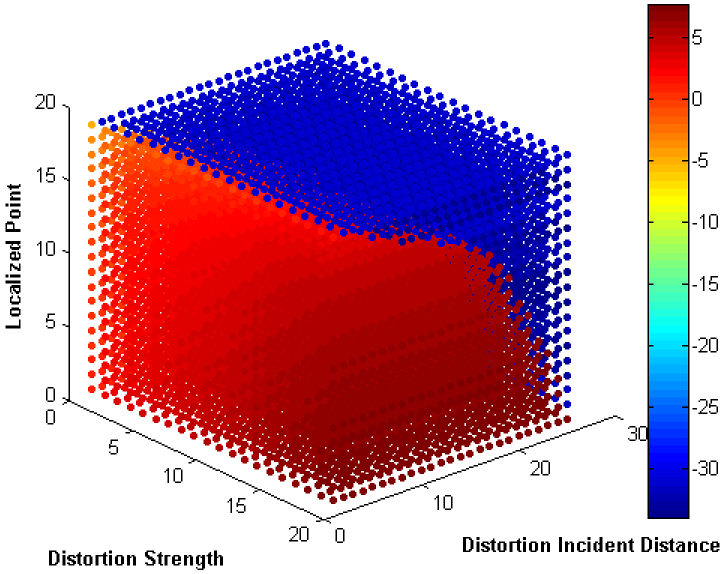


Figure 4.16: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: Rotated Trilateration: alternate view

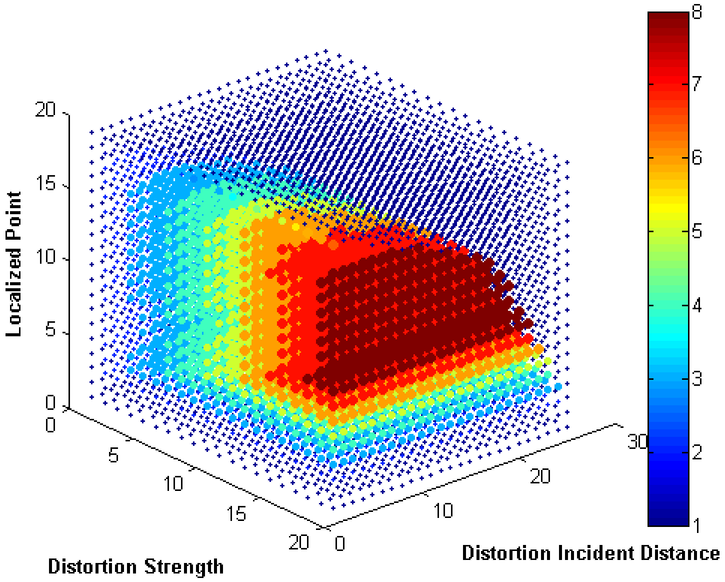


Figure 4.17: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: RADAR: alternate view

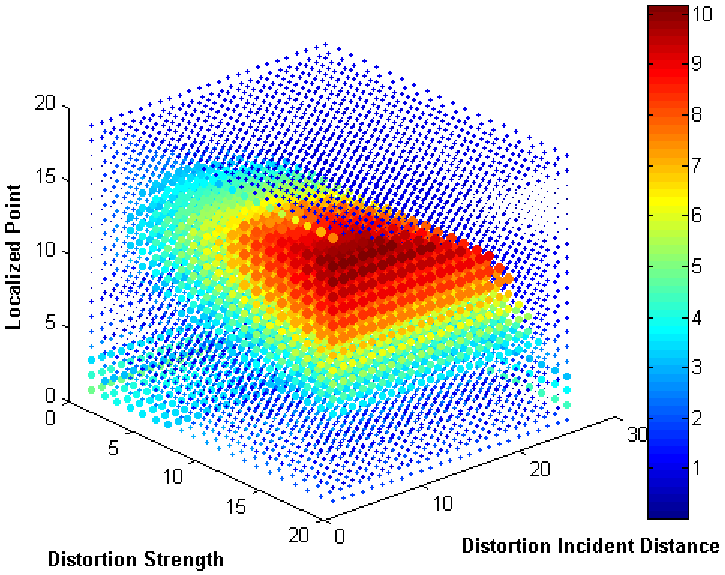


Figure 4.18: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: Area Based Positioning: alternate view

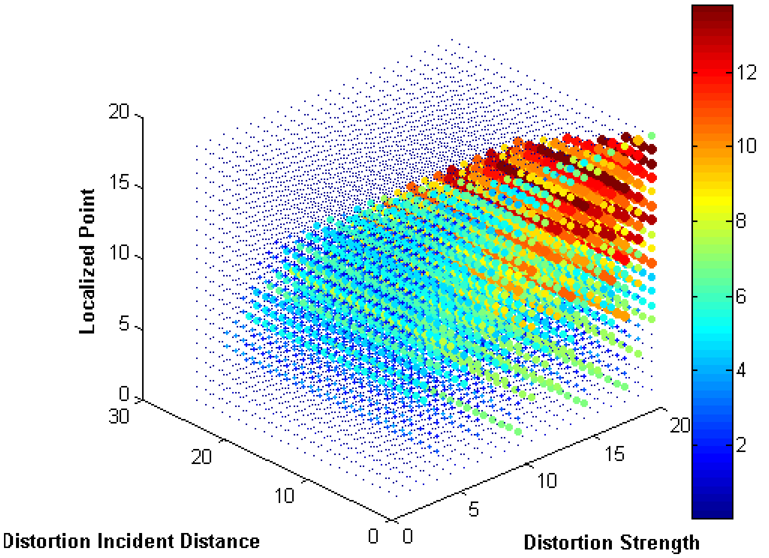


Figure 4.19: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: Simple Point Matching

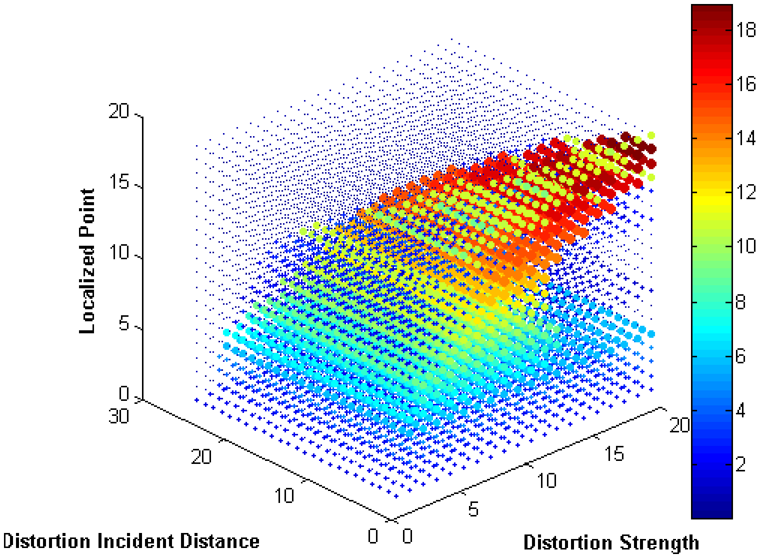


Figure 4.20: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: M1

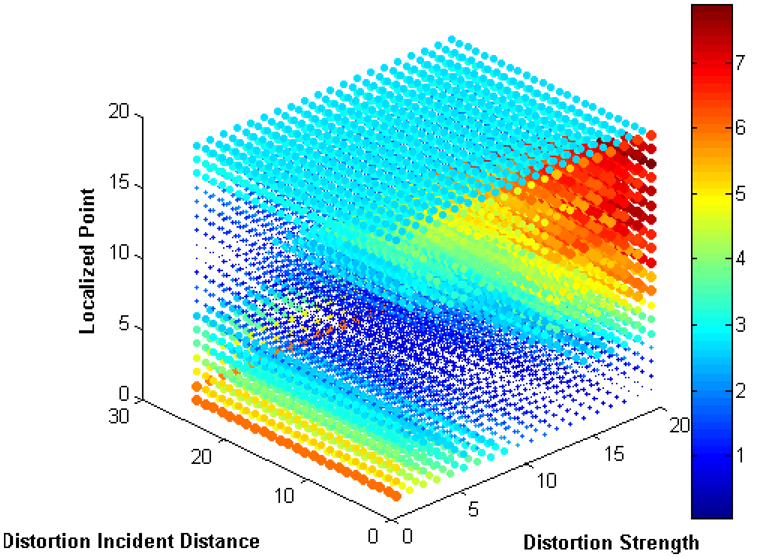


Figure 4.21: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: M2

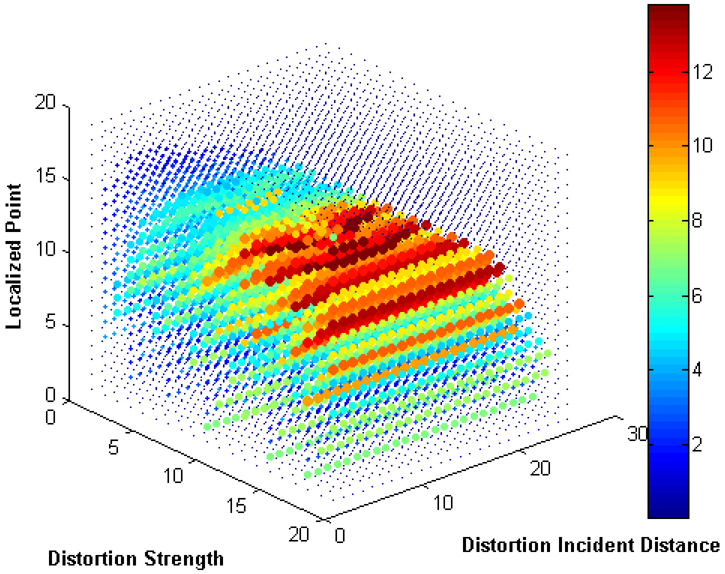


Figure 4.22: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: Simple Point Matching: alternate view

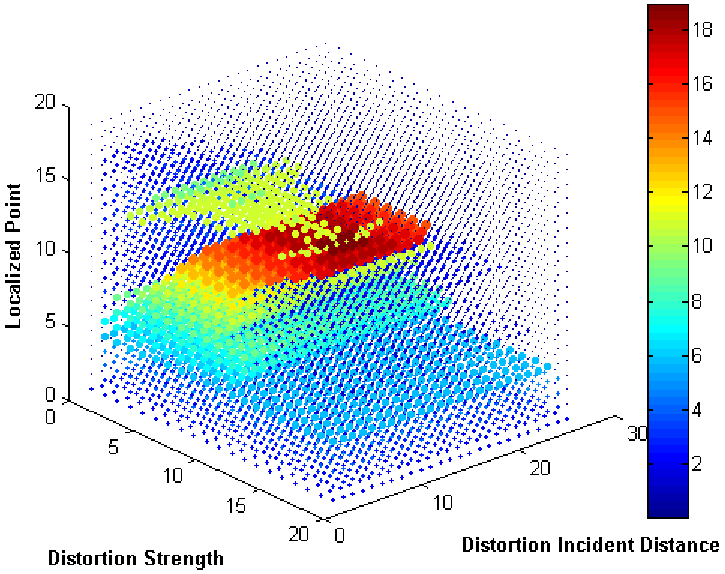


Figure 4.23: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: M1: alternate view

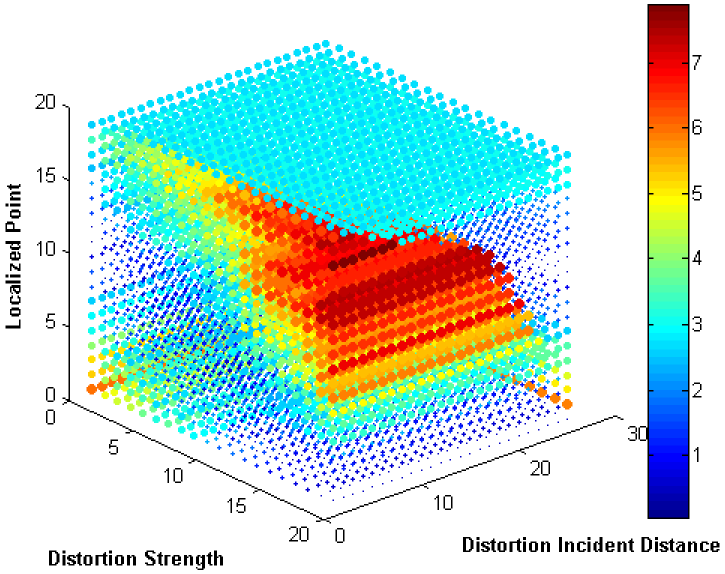


Figure 4.24: Localization Error per Descriptive Set Parameter per Point for Bias Distortions: M2: alternate view

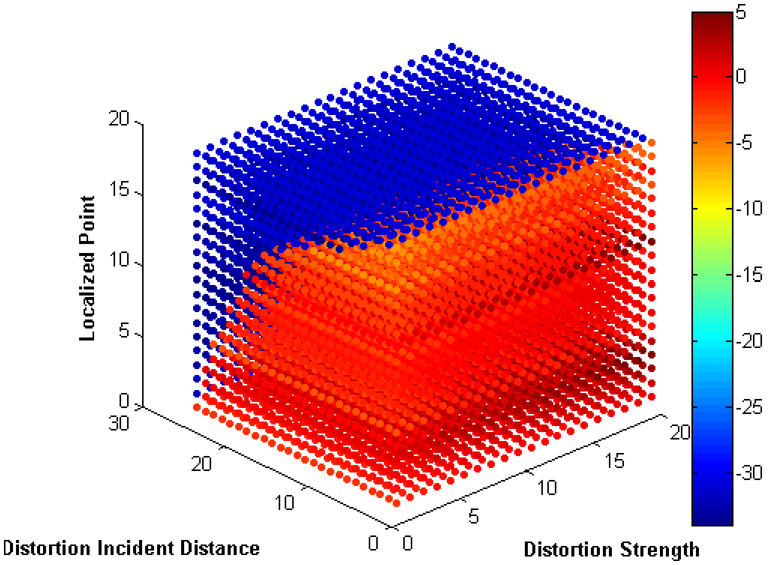


Figure 4.25: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: Rotated Trilateration

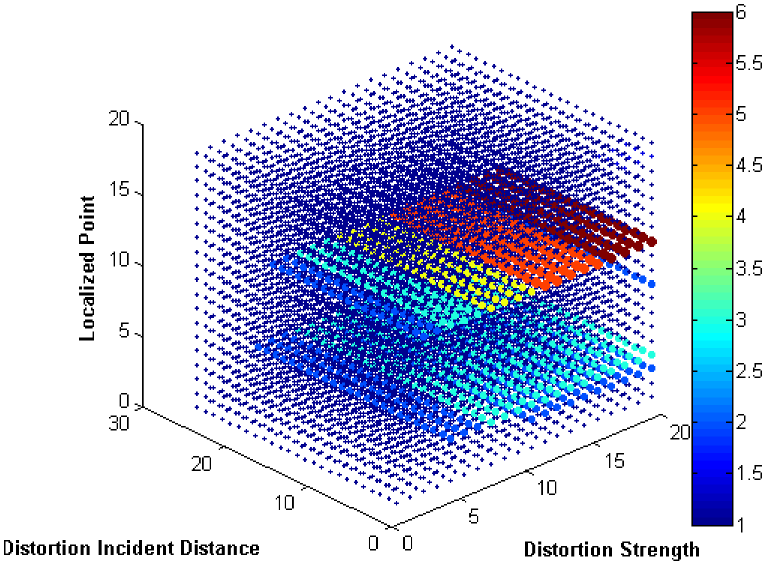


Figure 4.26: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: RADAR

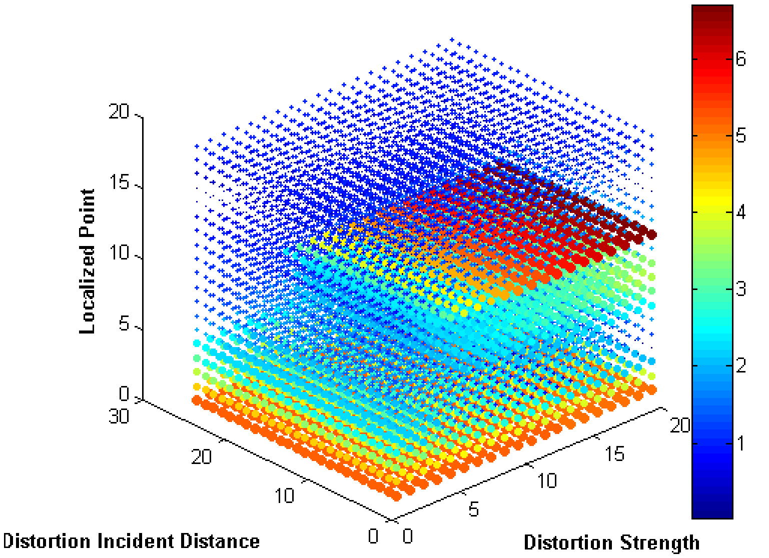


Figure 4.27: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: Area Based Positioning

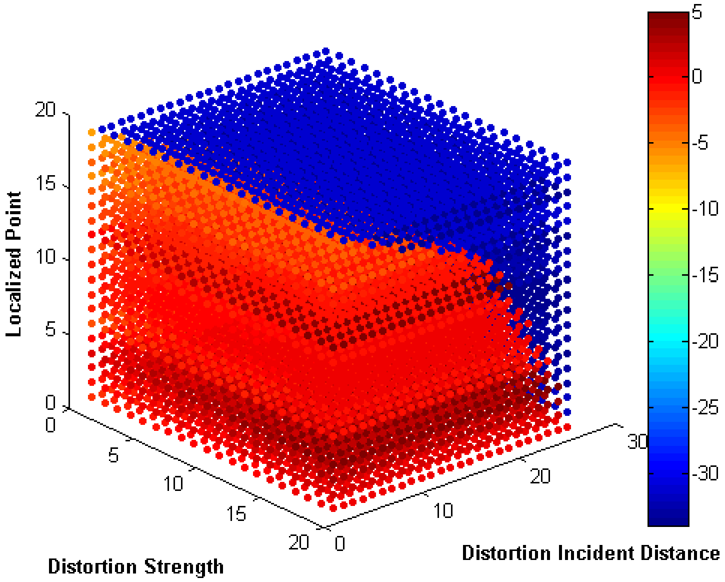


Figure 4.28: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: Rotated Trilateration: alternate view

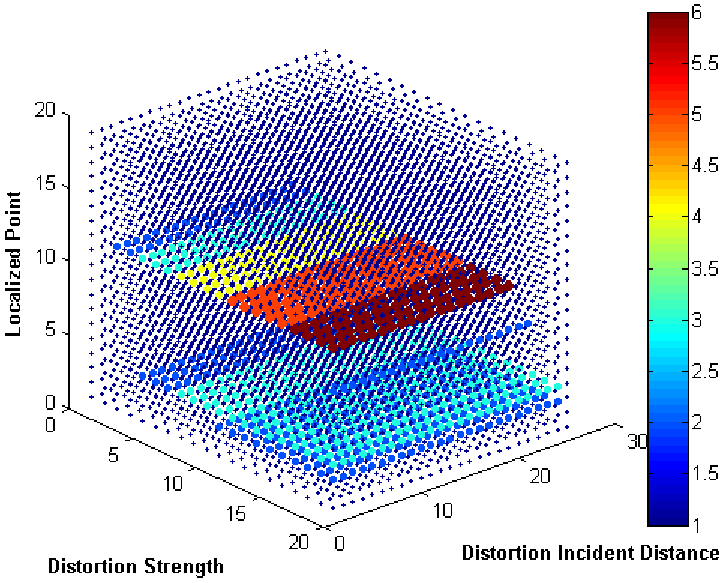


Figure 4.29: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: RADAR: alternate view

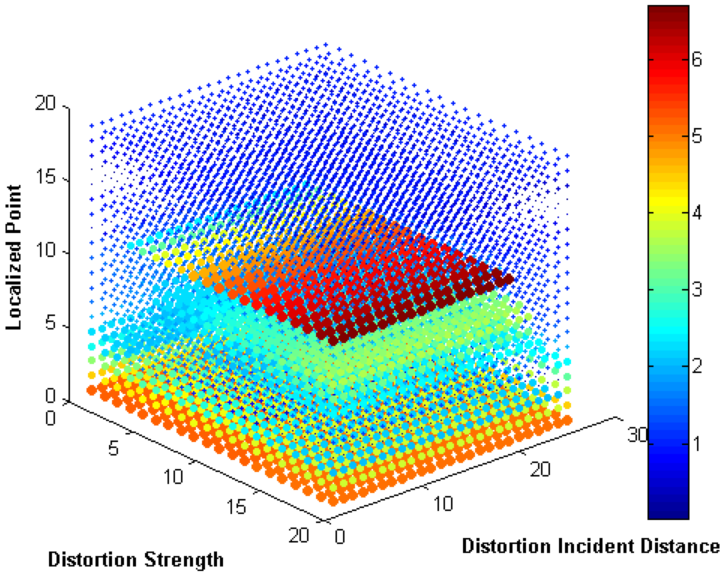


Figure 4.30: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: Area Based Positioning: alternate view

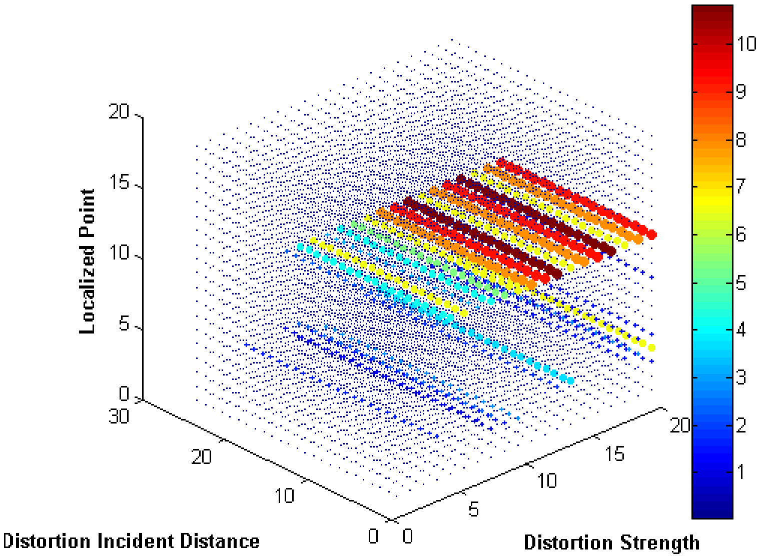


Figure 4.31: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: Simple Point Matching

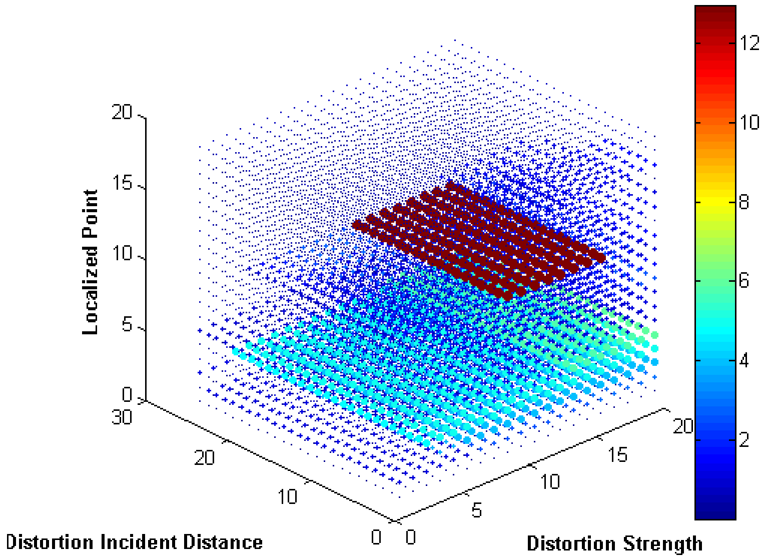


Figure 4.32: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: M1

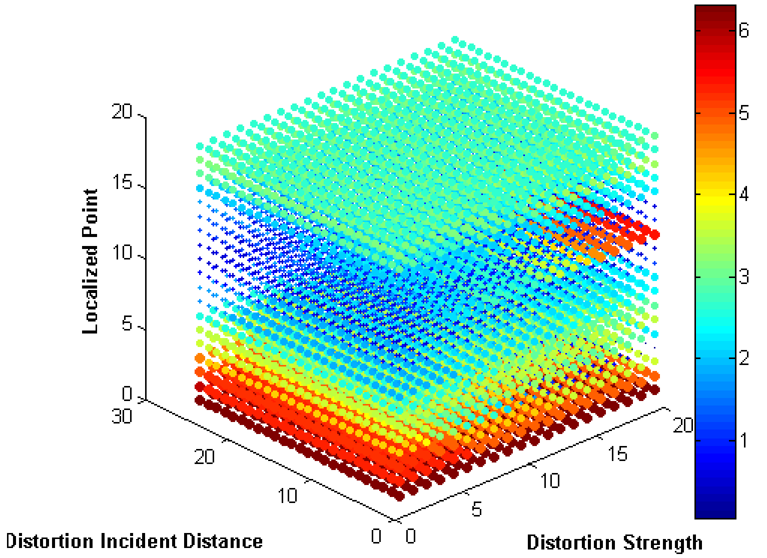


Figure 4.33: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: M2

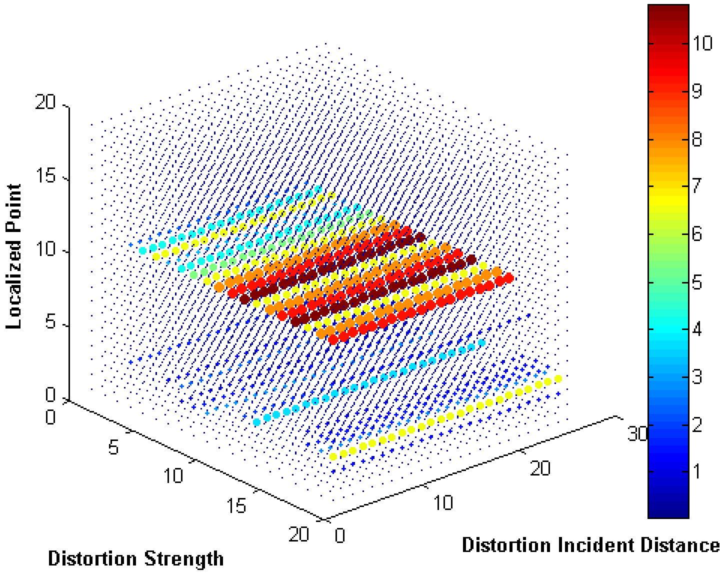


Figure 4.34: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: Simple Point Matching: alternate view

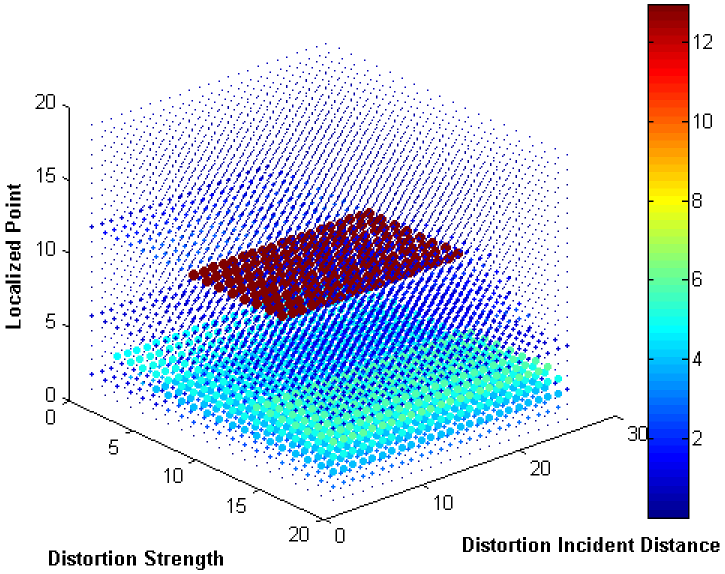


Figure 4.35: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: M1: alternate view

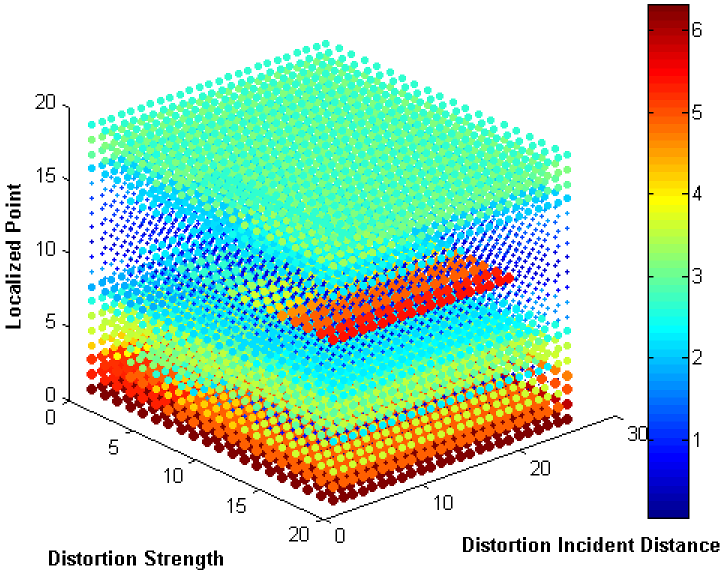


Figure 4.36: Localization Error per Descriptive Set Parameter per Point for Multipath Distortions: M2: alternate view

4.2.1 Analysis of Algorithm Behavior per Distortion Type

We examined the results for all algorithms over a given distortion type in order to determine if there are common behaviors or trends across all algorithms tested for each distortion type.

Attenuation

As can be seen in Figures 4.2 and 4.5, RADAR's error very steadily increases as more distortion strength is added and as incident distance decreases, causing more of the points to have attenuation applied to them. The only exceptions are the final three points, the farthest, point 20, in particular. Point 20 experiences at most 12 meters of error throughout all tests while the other points experience up to 16. This is likely because at 20 meters distant from the central LM, the 20th point is closer to the other two LMs that are experiencing no distortion, and the signal from the central LM is so low that even strong attenuations change its signal value very little.

ABP behaves much like RADAR in the presence of attenuation, which is to be expected. Although their methods of computation are quite different, they rely on the same fundamental principles. RADAR selects locations by numerical distance, while ABP calculates the most likely location by similarity to a distribution parameterized by the samples recorded. If the samples recorded are indeed diagnostic of the location, it stands to reason the most likely locations should be the locations whose signal values are most similar to the recorded samples. ABP's attenuation performance is more measured and gradual than RADAR's. ABP does have a much larger 6 to 7 meter region than RADAR, however its other error regions are smaller. ABP also has in particular a region of parameters it solves for exactly, even in the presence of error, resulting in a clear swath of very small dots in Figure 4.3.

M1 behaves quite a bit differently than RADAR and ABP, which is to be expected as it is a laterative algorithm, and ABP and RADAR are pointwise. M1's error performance seems to point to certain distinct configurations that cause it significant difficulty, which we can see from Figures 4.8 and 4.11. It appears that attenuations whose incident distances are approximately 10 meters from the point being localized cause M1 significant difficulty, as can be seen in the large error stripes for points 15 to 20 at incident distances 6 through 15 in Figure 4.11. This error stripe continues across all attenuation strengths, as is visible in Figure 4.8. This is all the more significant because we found that incident distance resulted in the smallest JSd sensitivity universally for the DDLHM metric. Even though the propagation mechanics would lead one to believe strength of distortion would be the most diagnostic of algorithmic error, it is not the case for the M1 laterative algorithm. Even more to the point, it seems the points just before and just after the troublesome areas result in some, but markedly less error. We can see in direct evidence the puzzling 'long tail' in the error of laterative localization algorithms.

The M2 and SPM algorithms elicited some very surprising behavior, each behaving unlike their algorithmic class. M2's error sensitivity to attenuations looks incredibly similar to ABP's. In particular, Figures 4.9 and 4.3 are extremely similar, down to the swath of exactly computed coordinates. By the same token, SPM's behavior in Figures 4.7 and 4.10 more directly compares to M1 than ABP or RADAR, even though SPM is a straightforward modification of RADAR. We believe these differences are due to the fact that SPM computes on interpolated data directly and M2 scores likely parameters.

Since the M1 algorithm statisically computes the best set of propagation parameters to describe training data that it later applies to translate the testing data into ranges direction, it is in effect interpolating. M1 doesn't necessarily compute its interpolation for the entire environment, but the degree of interpolation is not the issue. It suffices that the algorithm does parameterize a propagation model based on samples from the environment and applies it to data from an unknown point. The algorithm is based on the presumption that the lognormal model can properly describe each point's propagation. Likewise, even though the ABP algorithm does in fact apply interpolation to its data, ABP matches tiles of interpolated data and testing data numerically. Its relation is a Gaussian likelihood function rather than Euclidean Distance in signal space or Earth Mover's Distance, however method of reckoning aside, ABP computes a 'score' or 'goodness of fit' for each tile, rather than applying a model to convert signal to distance. Likewise, while M2 does compute parameters to a lognormal function to convert signals to distances, it does so by picking the set of parameters that best describe the training data. M2 runs through many random selections and picks the 'best' ones, in effect scoring all the possible interpretations of the collection of values it localizes. While it may not be evident from the algorithms' description of operation, the benchmarking process quite clearly demonstrates that the way the data is represented, as pointwise or laterative, has little to no bearing on algorithm performance. It is how locations are reckoned, either model-based translation or score-based comparison, that decides the general behavior and patterns of error with regard to attenuation distortions.

Bias

As can be seen in Figure 4.17, RADAR's error increases in stages as bias' distortion strength increases. It is however entirely insensitive to incident distance, as can be seen across the back of Figure 4.14. This stands to reasons since RADAR matches its results numerically. Given the meter-wide separation between the point in our synthetic environment, it should take some minimal amount of distortion to cause RADAR to shift the best-matching location from one point to another. Incident distance should have fairly little effect on RADAR since the distorted point will look more like other distorted points (which are close to the true location) than the others. RADAR also has the benefit of precise data from the two undistorted LMs, which should always indicate the correct point exactly, pushing it toward a correct conclusion.

ABP seems to handle bias distortions extremely well, resulting in a small, circular error wedge in Figure 4.15. Although its maximal error at 10 meters exceeds RADAR at 8, its overall error area is much smaller, resulting in a much smaller average error. ABP does however incur some error at very low distortion strengths, as can be seen for points 1 to 3 for distortion strengths 1 to 7 at all incident distances. These points are the points closest to the distorted LM and farthest from the exact LMs. This would cause the distorted LM to have a much stronger signal than the exact LMs. It seems that a small amount of bias is enough to push ABP to an incorrect location if it is applied to a close, strong signal without nearby correcting influences. This is direct consequence of its numeric reckoning. A small change to a strong signal can easily drown out the influence of exact, but weak signals.

In bias as well we see evidence of the reckoning method governing error sensitivity to distortion. As we can see in Figures 4.20 and 4.19, M1 and SPM look much more similar than SPM and ABP. Figure 4.22 depicts some interesting behavior of SPM. A series of stirations curve through distortion strength and are the same for all incident distances of bias for SPM. These are likely aliasing behavior. As we can see the error starts relatively low for a given point and strength, and as the strength and point localized more farther from the central LM error increases steadily, although not for all pairs of points and distortion strength. It is likely the bad points lie on a computational boundary between interpolated tiles and the clear areas of very low error in between do not and are able to tolerate significant error and still be localized with a fair amount of accuracy. M1's behavior is very similar to SPM. As can be seen in Figures 4.19 and 4.20, it has the same general shape and in Figures 4.22 and 4.23, the same degree. Although M1 does not have the same stirations through its error benchmark as SPM, the same general area, points 12 through 20 at bias strengths 10 through 20, cause them both significant difficulty.

M2's behavior resembles ABP's, although much less so than ABP and SPM. In Figures 4.24 and 4.18, we can see that ABP and M2 both have their highest errors mostly confined to the strongest distortion strength and the farthest points, while M1 and SPM both have error trends that increase directly as the bias strength increases. M2 and ABP both are much less sensitive to the point selected and are much more sensitive to the strength of the bias distortion. M1 and SPM seem to have difficulty only with certain points over a wide range of distortion strengths. It stands to reason that since the SPM and M1 use model-based translation of data, that certain points would result in configurations that are more fragile and sensitive to distortion. ABP and M2, which make conclusions about data based on numeric comparison, are fairly agnostic to where in the environment the point is, and are very sensitive to the degree of distortion, which would confuse their comparisons. ABP and M2 also have the same exception to point location governing error sensitivity in their similar early error strips over the lowest distortion strengths and only for the closest points, clearly visible in 4.15 and 4.21.

Multipath

RADAR handles mutipath fairly well, except for one point. As can be seen in Figure 4.26, RADAR does incur some error at points 3 and 4, although the one that causes the most error by far is point 12. This point roughly corresponds with a signal null, as does point 4. At all other points, multipath does not affect the propagation overmuch, and results in very low error. SPM and M1 have very similar behavior, for much the same reason, as can be seen in Figures 4.31 and 4.32. Beyond the deep signal nulls, multipath has fairly little effect on the algorithms that use model-based data translation since, in most cases, the model is correct. ABP and M2 however with their score-based comparisons have some difficulty with multipath with their score-based comparisons. Since multipath propagation causes both signal peaks as well as troughs and nulls, there are many more features that can cause numeric comparisons to come up with unexpectedly different values than the model-based translations. As such, ABP and M2 both have similar benchmarks, as can be seen in Figures 4.27 and 4.33.

4.2.2 Analysis of Algorithm Behavior across Distortion Type

We examined the results of each algorithm across all distortion types to determine if there are any common factors or trends in error performance that are general enough to hold given any distortion.

RADAR

Across all distortion types, RADAR has fairly different sensitivity characteristics. When localizing in the present of attenuation, it seems to perform reasonably well, incurring no more than 10 meters of error until experiencing strong distortions over the last few points. It does however have a much lower error floor, incurring at least 2 meters of error for nearly all attenuation parameterizations, as can be seen in Figure 4.2. RADAR's difficulty with late-edge distortions continues into bias, where lateration point and distortion strength are much more stable predictors of error than incident distannce. Unlike its attenuation results however, RADAR has can localize farther points more accurately when dealing with bias. As can be seen in Figure 4.14, RADAR can localize all points rather well until distortion power increases to the 15th reduced set bias strength parameter and above. RADAR is surprisingly resilient to multipath distortions. It sensitivity to localized point is less remarkable due to the signal null behavior of multipath. In Figure 4.26 we can see that the first null point causes it some difficulty, resulting in approximately 2.75 meters of error. The second null point however see RADAR gracefully degrading across distortion strength thresholds. Multipath strengths 3, 7, 11 and 17 all cause at least a 1 meter increase in error. From this we can conclude RADAR can be expected to localize points up to 10 meters away with fairly little error no matter the strength of attenuation, only up to 5 meters away or with less than a 15dbm drop when dealing with bias, or with less than 30% multipath.

ABP

ABP has a very charactertistic shape to its error performance for both attenuative and bias distortion types, as we can see in Figures 4.3 and 4.15. Its error has a particularly ovoid shape, the greatest error centered on a distant point at strong distortion. ABP does however have an oddly distinctive error bump at very low distortion strengths for very close points. The smoothness and graceful degradation across all distortion types carries over to multipath as well, where ABP's reaction to the deep second signal null is fairly smooth, as seen in Figure 4.27, in sharp contrast to the threshold behavior of RADAR. This does not however recommend ABP to multipath distortions, as this graceful degradation causes it localize with greater error across many multipath scenarios that even an extremely simple algorithm, like RADAR, can handle easily. The odd low-power error bump can also been seen as a very large 5.3 meters of error for the first point localized, regardless of the multipath strength or incident distance. From this we can see that ABP is often quite resilient to both attenuation and bias, with an all but identical error benchmark per parameter, although it is particularly susceptible to multipath. In particular, ABP's difficulty localizing points close to a LM causes additional error in all cases, making it apparent that ABP should be be used to localize any point closer than 3 meters to an LM.

SPM

SPM, like ABP, handles both attenuation and bias similarly, although unlike ABP, it does not handle the distortions in a particularly graceful. As can be seen in Figures 4.7 and 4.19, the general shape and distribution of error is quite similar for both distortions. The overall shape of the benchmarking error is also somewhat ovoid, although it is not centered on or consist of cohesive regions of error.

SPM instead experiences a given amount of error based on point localized and distortion strength, but only within a given incident distance bound. Beyond the bound, the error sharply decreases, or disappears entirely. The generally ovoid shape of both SPM and ABP may be an artifact of the interpolation that both use, while the cohesion of error regions is likely caused by their reckoning method, as discussed above. SPM is unlike ABP in that it handles multipath extremely well, with only a few configurations causing trouble around the first signal null and alternating strips of error per multipath strength along the second, as can be seen in Figure 4.31.

M1

The laterative algorithms continue their trend of mirroring the performance of one of the two pointwise methods between distortion types. M1 handles attenuation much better than attenuation, with much less overall error, as can be see in Figure 4.8. The critical difference here may be interpolation. SPM and ABP both have ovoid error regions across distortion types (to varying degrees of coherence), and while the curve and degree of M1's error benchmark is quite similar to the outer edge of SPM's performance, along the distant points to be localized, M1 does not have the same error spread along close points at high attenuation values. In fact, M1 handles very strong attenuations quite well, with very little error past the 15th attenuative strength parameter of the reduced set. This trend of M1 continues through bias, with it performing quite well at high bias strengths, except for distant points to localize, as can be seen in Figure 4.20. From its behavior over attenuation and bias, it seems M1 has a certain error floor. At lower distortion strengths and at closer points to localize, M1 isn't as picky and spreads varying amounts of error out over wide regions. As the distortion strengths increase and the points to localize get farther away, M1 compresses the same amount of error into fewer configurations with much greater error. It seems that the 10 to 15 meter area is exceptionally fraught with very strong errors when localizing distant points. M1 also handles multipath fairly well, much like SPM. As can be seen in Figure 4.32, M1 handles the first signal with middling error of about 5.2 meters, but is handles the second null quite well, localizing with very little error until the multipth strength increases past the 9th parameter. From this we can conclude localizing points closer than 10 meters with M1 should be fairly error-free, or points past 15 meters that contain very little multipath. Otherwise, strong error regions begin to monopolize the benchmark area.

M2

M2 behaves very much like ABP, with an all but identical error performance for attenuation, as can be seen in Figure 4.9, with the same peculiar ovoid error region, the same clear region below and a similar error segment over low-strength attenuation on close points. M2 has fairly different performance for bias however, looking like a cross between RADAR and ABP. As we can see in Figure 4.21, M2's benchmark has a fairly rhomboid shape, describing several regions that are bounded fairly linearly along both bias strength and localized point. For nearly every 1 dBm increase in bias, the error function seems to shift to a point 1 meter closer. This behavior marks out some dependency within the M2 algorithm that can hopefully be tuned out, as it seems to be rather regular and decidable. Out of all the algorithms, M2 handles multipath the worst, although with very different error behaviors from the other algorithms. As we can see in Figure 4.33, M2 actually does not have much trouble with the second signal null in particular, with a much lower maximal error and average error at that point than any other algorithm. It does however seem to have ABP's same difficulty with low multipath strengths applied to close points, but to an extreme degree, resulting in errors above 6 meters for very little multipath. From this we can conclude M2 has some type of algorithmic defect that is exposed when localizing with bias disortions, that it should localize with little error in the presence of attenuation if the attenuation is somewhat weak, and that it can handle fairly strong multipath well, so long as it is localizing points more than 5 meters from the LM.

4.3 Closed-Form Algorithmic Benchmarking

An algorithm's error results when computing on the synthetically distorted signal values generated to exercise them fully for benchmarking purposes provide a practical relation between distortion characteristics and localization error. These results however were generated purposefully and have no larger context. While the test suites may indicate an algorithm is sensitive to a certain type of distortion or parameters, it is not clear how much additional resolution is to be gained by improving its handling of that particular feature. Since we have only compared algorithms' benchmarks against other algorithms' benchmarks, we haven't any way to judge the value or capability of a given algorithm's error characteristics beyond our chosen test suite. It could very well be the case that all the algorithms we tested with have a particular flaw that cause a certain result that is not a direct consequence of the distortion types and parameterizations. In order to regard our benchmarking results as diagnostic rather than as a heuristic, we must determine the error-causing capability of each distortion configuration tested in a direct manner that does not in any way reduce any of the error. Moreover, while when benchmarking we have the distinct luxury of working with known amounts and types of distortion. When examining localization results computed on live data we have no such assurance. Much to the point, if we knew the precise amounts and types of distortions applied to live data, given the benchmarking results, we could correct for them. What is necessary is a way to gauge how well an algorithm could compute on any signal data, with known or unknown distortion parameterization.

An algorithm's capacity to localize can be further judged by comparing it to the performance of rotated trilateration; a laterative localization algorithm with a closed-form error expression, directly linking ranging error with lateration error by geometrically computing a location based on ranging alone. The resultant model provides a ready profile of how strongly different distortions directly affect a particular laterative engine. While the behavior of rotated trilateration is not necessarily generalizible to all algorithms, it is comparable in that it localizes as well. By comparing a given algorithm's performance metrics to rotated trilateration, especially for corresponding benchmarking parameters or over the same live data, it can supply a ready context to judge algorithmic efficacy. In particular, no algorithm should ever perform worse than rotated trilateration, as it has absolutely no error reduction capabilities. The degree to which a given algorithm improves upon the results of rotated trilateration can act as a judge of its capability.

4.3.1 Definition of Rotated Trilateration

Assuming that there are no distortions on sampled RSSIs, it is a simple matter to translate signal values to ranges. The remaining difficulty is that we have only ranges from LMs, not the direction from which the signal came. This leaves us with a radius per LM describing potential transmitter locations about its circumference. If we again presume there is no error or distortion, then each circle should coincide at one precise point. Admitting the possibility that the ranges may not be exact however, we need a method that can directly compute the best possible point of intersection that is closed-form and not statistical or error-reducing. If the ranges do not align on a single point based on the locations of the LMs that recorded the signal values that we translated, we will devise a coordinate system where we can not help but compute a proportionally correct position. We will take one LM to be origin, and the distance from the origin LM to another as displacement along the x axis of our new coordinate system. We will normalize that distance and consider it to be our unit vector. We will then consider the last LM to lie on only the y axis of our new coordinate system, and will compute its distance and coordinates in terms of our new unit vector. Once we translate the ranges we computed into our new coordinate system, we can not help but compute their intersection since each LM now lies along only one axis of our new 2-dimensional space. Since we have three range measurements in two dimensions, we can calculate an exact solution for the location. This location is computed in the new coordinate space, so once it is rotated back into the original coordinate system, we have an exact localization computed in closed form built on the presumption that ranging based on the lognormal equation is correct. No error is reduced and no statistical estimations are applied. The same error in ranging will have the exact same error in location calculation every time.

Let LM_0 be the coordinates of Landmark 0
Let LM_1 be the coordinates of Landmark 1
Let LM_2 be the coordinates of Landmark 2

Let $RSSI_0$ be the RSSI sampled at Landmark 0 from the transmitter to be localized
Let $RSSI_1$ be the RSSI sampled at Landmark 1 from the transmitter to be localized
Let $RSSI_2$ be the RSSI sampled at Landmark 2 from the transmitter to be localized

Let r_0 be the range from Landmark 0 to the transmitter to be localized
Let r_1 be the range from Landmark 1 to the transmitter to be localized
Let r_2 be the range from Landmark 2 to the transmitter to be localized

Compute ranges from RSSIs:

$r_i = \sqrt{\frac{0.1230^2}{16*\pi^2} * \frac{1}{10^{\frac{RSSI_i}{10}}}}$

Compute distance between LM_1 and LM_0:

$d_0 = \|LM_1 - LM_0\|_2$

Compute relative deviation of $LM_2 from LM_0$:

$d_1 = LM_2 - LM_0$

Determine magnitude of directed difference between LM_1 and LM_0 relative to its norm, computing the sensitivity of coordinates to distance in the space:

$e_x = \frac{LM_1 - LM_0}{d_0}$

Standardize the deviation space by multiplying each dimension of the deviation of LM_2 from LM_0 by the ratios of coordinate to distance sensitivity in the space:

$i = ex \cdot d_1$

Compute the relative ratiometric sensitivity between the dimensional deviation sensitivity of the LM_2,LM_0 leg and the LM_1,LM_0 leg:

$e_y = \frac{(d_1 - i * e_x)}{\|(d_1 - i * e_x)\|_2}$

Multiply each dimension of the relative difference of LM_2 from LM_0 by the normed relative sensitivity ratios between the LM_2,LM_0 leg and the LM_1,LM_0 leg:
$j = e_y \cdot d_1$

Compute the vector orthonormal to both the relative sensitivities of the LM_2,LM_0 leg and the LM_1,LM_0 leg, forming an orthonormal basis:
$e_z = e_x \times e_y$

Let e_x be distance along the x axis in our new basis
Let e_y be distance along the y axis in our new basis
Let e_z be distance along the z axis in our new basis

We have now translated our ranges into a new coordinate system where each range lies along one axis only by rotating the coordinates of the landmarks, zeroing out constraints that cause too many dependencies to compute the location directly in closed form.

Compute the location of the transmitter in the relative coordinate system and rotate back into the original one:
$x_{rotated} = \frac{\left(r_0^2 - r_1^2 + d_0^2\right)}{2 * d_0}$
$y_{rotated} = \frac{\left(r_0^2 - r_2^2 + i^2 + j^2\right)}{2 * j} - \left(x * \frac{i}{j}\right)$
$z_{rotated} = \sqrt{r_0^2 - x^2 - y^2}$
$LocalizedCoordinates = LM_0 + x * e_x + y * e_y + z * e_z$

4.3.2 Rotated Trilateration Error Characteristics

We computed the same benchmarks for the Rotated Trilateration (RT) algorithm that we did for each of the other algorithms we examined above. In many cases the errors were so large, that in order to represent them compactly, we graphed only the logarithm of the error, base e. RT did experience some small error in non-distorted cases due to the fact that even though our LMs coordinates are in 3-space, they are same along the z axis, placing them essentially on a plane. When rotating the computed coordinates back, very often the computation does not solve to exactly 0 along that single axis, likely due to roundoff error in the multiple square roots. Even so, the small amount of error induced by the floating z axis parameter was in all cases extraordinarily negligable in contrast the massive error caused by even the smallest amount of distortion.

As can be seen in Figures 4.1 and 4.4, RT experienced essentially no error for non-distorted cases and gradually increasing error as both distortion strength and localization point and, to a degree, even incident distance increase. It may seem from the heatmap that RT's error when dealing with strong attenuations far from the transmitter are lower than ABP's or RADAR's, from their very dark regions in Figures 4.3 and 4.2. This is not the case since the error value graphed in the RT benchmark blocks is the logarithm of the algorithmic error, which in some cases hundreds of meters greater than the 16 or 15 meters of maximal error for RADAR or ABP. In Figures 4.13 and 4.16 we can see that RT handles bias type distortions much more regularly than it does attenuations. While RT does perform better on some attenuation parameterizations than bias, some of the errors caused by attenuations are incredibly large. RT handles multipath distortions in a similar manner, as we can see in Figures 4.25 and 4.28. Error increases around the points that cause signal nulls, however otherwise multipath performance strongly resembles bias performance.

These results offer an interesting perspective. In a pure sense, attenuations cause incredibly high error values if the lognormal model is taken as directly correct, while bias results in much more homogenous error. The other algorithms benchmarked reacted in some part similarly, in part not. In all cases, for all algorithms, maximal error for bias was lower than that for attenuation, indicating that, in general, any localization algorithm should handle bias better than attenuation. There was difference in degree, and M1 in particular had an incredibly modest reduction in maximal error between Figures 4.8 and 4.20, although it was a reduction. From these examples and the performance of RT, we can conclude any localization algorithm that benchmarks with higher error in bias than attenuation is experiencing some odd effect that is likely an artifact of its computation. Another lesson we can draw is, comparatively, how well localization algorithms handle attenuations. Attenuation distortions caused extremely high error for rotated trilateration, however no algorithm had error anywhere near the extraordinary errors RT experiences. For multipath distortions we can see that RT is a bit more indicative of the general trends. Maximal error is roughly the same and RT has increased error around the null points, as do the other algorithms. One marked difference however is that RT experiences fairly high error throughout all distorted configurations, while only M2 comes close to the same performance, with any appreciable error outside of the null regions, as can be seen in Figure 4.33.

From the above we can see that Rotated Trilateration provides a deterministic context between algorithms by establishing a lowest common denominator. No algorithm performs as badly as RT over any benchmark parameters. RT also illustrates some of the error trends similar in the other algorithms: the multipath nulls, the reduction in maximal error from attenuation to bias and again to multipath. RT also is useful in that it demonstrates just how much error attenuation can cause if it is not correctly handled. While such a context is not as directly necessary when comparing algorithmic error across known parameters in a synthetic environment, it will become critical for establishing performance expectations when computing on live data where the exact distortion parameters are unknowable.

Chapter 5
Localization Environment Analysis

While we have developed a method to gauge localization performance that is algorithmically agnostic, there remains no method that is environmentally agnostic. In order to assess algorithms under the influence of a precise amount of distortion we removed the unpredictability of the environment by computing all tests in a synthetic one. The common method to deal with environmental distortion is to test an algorithm in multiple environments and try to establish general trends in the error. If all environments that are tested in result in similar localization error characteristics, it is reasoned that the algorithm will have similar behavior in most environments. While this is certainly a reasonable strategy, it is particularly fraught since the heterogeneity of signal effects, coupled with potentially prohibitive set up and deployment times, makes sampling enough environments to be diagnostic extremely unlikely. Due to these logistical difficulties, most algorithms are tested on a small group of environments, if any more than one. While testing an algorithm in more than one environment can grant additional perspective to its performance, such tests are far from diagnostic or conclusive.

Averaging localization results over several environments together or examining them in concert under the assumption that they communicate qualitatively similar results that can be directly compared is not sound. While all localization algorithms tested in an environment will all result in some degree of error, those results are inseparably tied to the data which came from that particular environment. Every signal strength recorded at a particular location in an environment is the result of innumerable distinct propagation paths coinciding. Any movement to a new location is another twist of the environmental kaleidoscope, with a distinct set of reflections, refractions, absorptions and cancellations combining to create an entirely new gestalt effect. Likening any point to any other devoid of a model to determine how these multiple effects change is an exercise in heuristics, at best. Given that each point in any single environment experiences a set of signal effects distinct from each other, attempting to compare localization results over a group of such points is unlikely to result in any useful or distinct information. The problem with such techniques is that they do not address the fundamental issue: what about the environment causes a given result?

It is not possible to determine from a single signal value what caused it. Any number of distortion configurations could result in a given signal strength just as an infinite number of pairs of numbers could sum to 12. In order to determine why an algorithm performs as it does in a given environment it is necessary to determine what type of difficulties the environment poses. Since we demonstrated that the DDLHM can be used to identify and determine the type and parameters of common signal distortions in 3 and that we can benchmark algorithm performance in order to determine how they react to a given distortion type and parameterization, all that remains is to exploit both analyses in order to determine the dominant type and parameters of signal distortions in the environment and their expected effects. This process would extend our capabilities considerably. Firstly, we could now determine what particular aspects of an environment would likely cause the maximal types of localization error. Rather than making guesses based on conjecture and heuristics, equipped with an algorithmic benchmark and an environmental assessment, it is possible to determine what types of distortions are causing the worst of algorithmic error. Secondly, it would allow us to determine an error expectation per environment and algorithm. As improvements are made, the algorithm could be re-assessed, allowing reasoned, rational, exact and duplicatible development of localization algorithms. Finally, we could generate an general expected degree of localization difficulty a-priori by assessing the performance of the closed-form Rotated Trilateration algorithm, which would also provide us with a common context in which to evaluate other algorithms, a-posteriori.

5.1 Environment Assessment

Localization is fundamentally a translation from a topological space to a metric space. Physical space is metric; the distance between two points is a reflexive, constant quantity and any location's coordinates can be precisely measured to any granularity larger than Planck lengths. The indoor signal space is topological; the signal strength may change between two points over time, might not be reflexive, and signal strength may stay the same over a given region, enlarging 'points' of the same signal strength to cover a region. It is a simple matter to translate from a metric to a topological space since a metric space is more strongly defined. The process of localization is the opposite transformation; translating a series of measurements in the less well-defined topological signal space into a precise set of coordinates in the metric physical space. If not for the distortion manifold of the environment, radio signals would propagate precisely as described by the lognormal model and the translation from signal space to physical space would be fairly error free. The act of assessing the error-causing capacity of an environment is then the process of determining the properties of this distortion manifold, and constructing an error model.

A common argument is that in order to determine the expected error when localizing some object based on its signal data, the error model would have to determine the likelihood of the organization, materials, and disposition of people and objects in the environment, since they all affect the radio signal received and therefore the localization result. Such an error model is patently impossible to construct. While it is true that all of these factors do affect what happens to signals in the environment, it is not our intention to determine the floorplan of the environment, but to assess the degree to which the environment distorts our expectations of lognormal propagation. Predicting the precise disposition of an environment is not possible, however determining how well our expectations fit it is.

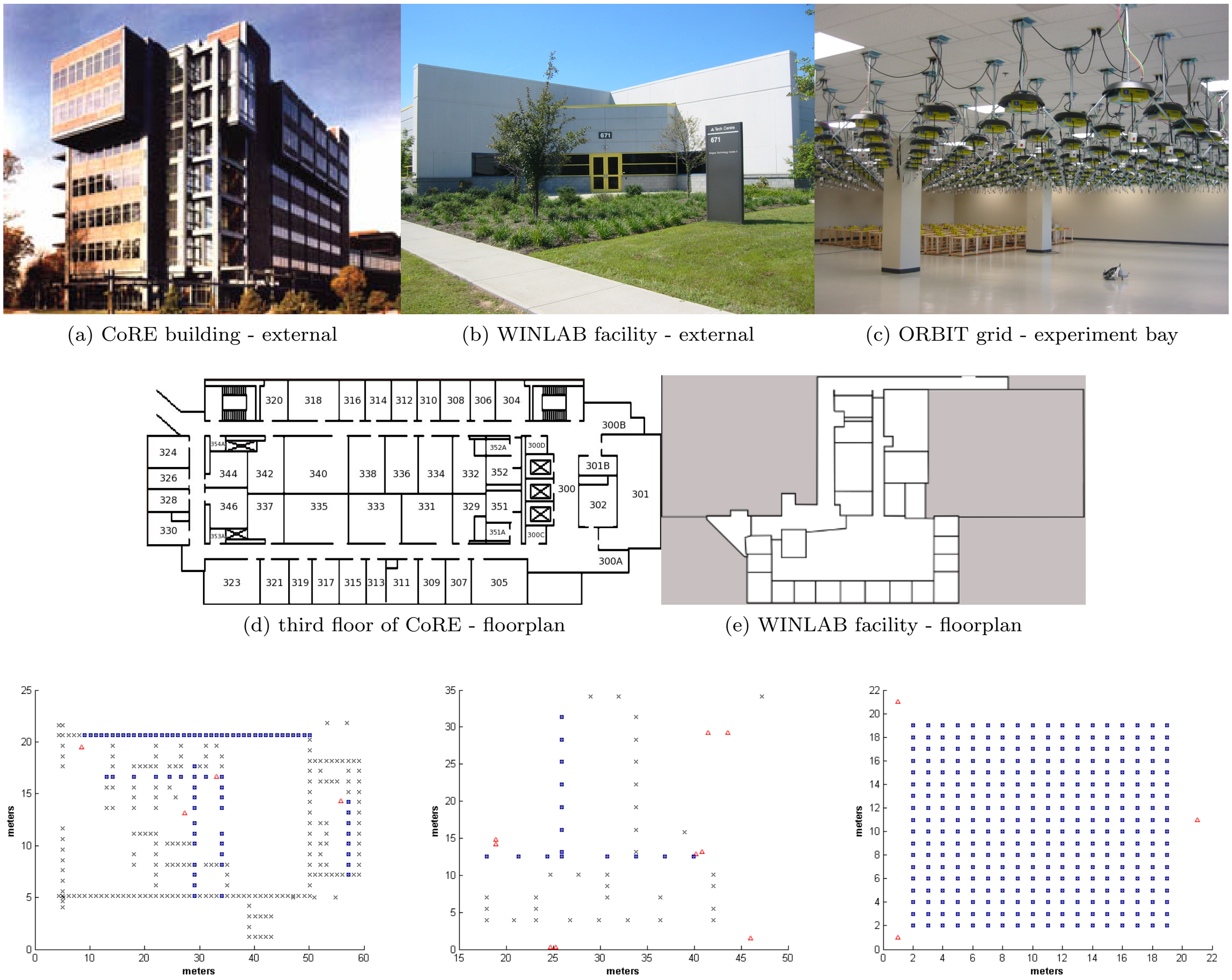
The key to doing so is judicious use of the DDLHM. As we demonstrated in Chapter 3, we can determine the dominant distortion type and parameters present in a series of signal samples taken along a straight line extending from a LM. By seeking such signal vectors in an actual environment we can determine the characteristics of distortions applied to them. Not all environments are constructed of straightline paths, however we will use only that data that we can and will presume the paths we have are indicative of the environment. Since people often tend to collect data along hallways and walkways, nearly all environments have fairly long vectors of signal data. So long as there is an LM near one of the ends, the data can be used to assess distortion characteristcs.

5.1.1 Environments Sampled

The Core environment is the third floor of the Computing Research and Education building on the Rutgers University campus, consisting of several academic computer laboratories and offices arranged around a rectangular arrangement of four hallways, as can be seen in Figures 5.1a and 5.1d. The labs are filled with various electronic and computer equipment, metal shelving and wall-mounted electrical conduit. Core has 6 collections of sampled points that are amenable to analysis since its radio architecture is built to enhance coverage for the general 802.11 communication network. The LM locations are represented by red points, locations of radio samples used for distortion vector analysis as blue points, and all other data collected as black points in Figure 5.1f.

The WINLAB environment is the main working area of the Rutgers University Wireless Netwoking Laboratory, consisting of a large area of half-height cubilces, glass-fronted offices, as well as storage and service rooms containing electrical and computer equipment, as can be seen in Figures 5.1b and 5.1e. WINLAB has 6 sample paths as well. Unlike CoRE, WINLAB was instrumented particularly for localization testing, so its paths extend for quite a long segment of the environment, as can be seen in Figure 5.1g.

The Grid environment consists of the Orbit computing grid lab in WINLAB, as can be seen in Figures 5.1c and 5.1h. The Orbit grid consists of a 20 by 20 meter square of 400 single-board ITX form-factor computers suspended from the ceiling. Each computer is one meter away from its



(a) CoRE building - external

(b) WINLAB facility - external

(c) ORBIT grid - experiment bay

(d) third floor of CoRE - floorplan

(e) WINLAB facility - floorplan

(f) CoRE Test Vectors

(g) WINLAB Test Vectors

(h) Grid Test Vectors

Figure 5.1: Areas and environments sampled

cardinal adjacent neighbors and and has two wireless cards. Since every single point has two wireless cards, there is no need for a specific machine to be an LM. Since any of the wireless cards could receive signals and any other broadcast, we recorded on nearly every row and column, resulting in 18 horizontal and 18 vertical paths.

5.1.2 Distortion Characteristics of the Core Environment

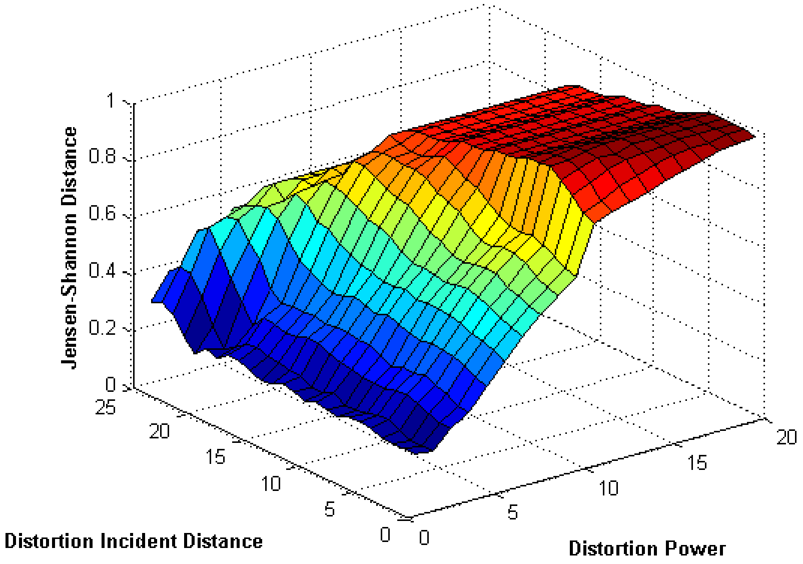


Figure 5.2: Core Path 1 DDLHM JSds: Attenuation

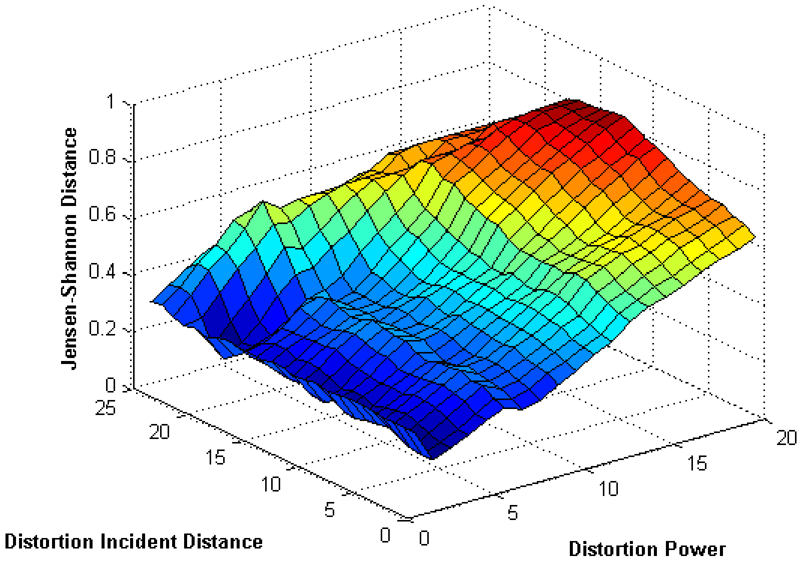


Figure 5.3: Core Path 1 DDLHM JSds: Bias

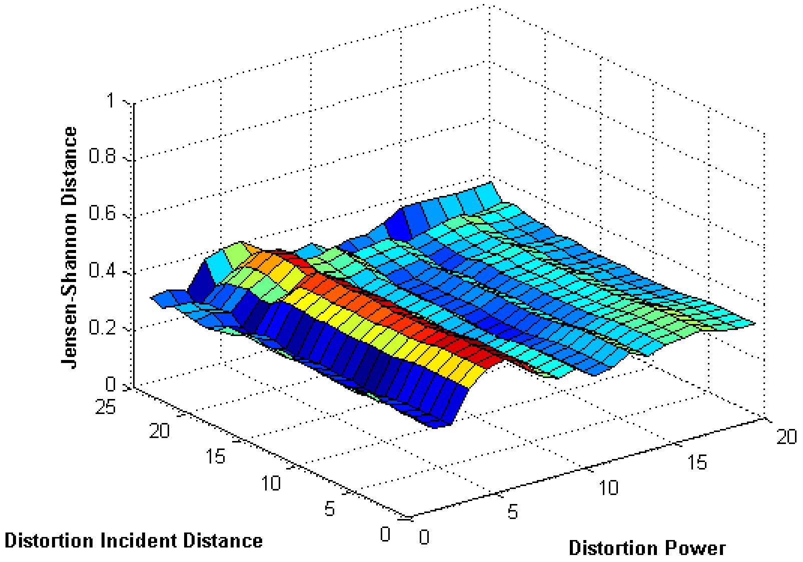


Figure 5.4: Core Path 1 DDLHM JSds: Multipath

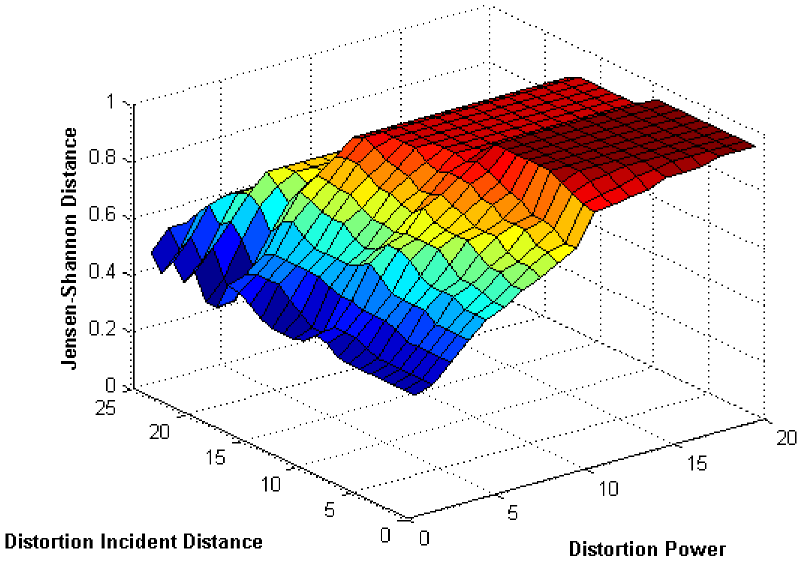


Figure 5.5: Core Path 2 DDLHM JSds: Attenuation

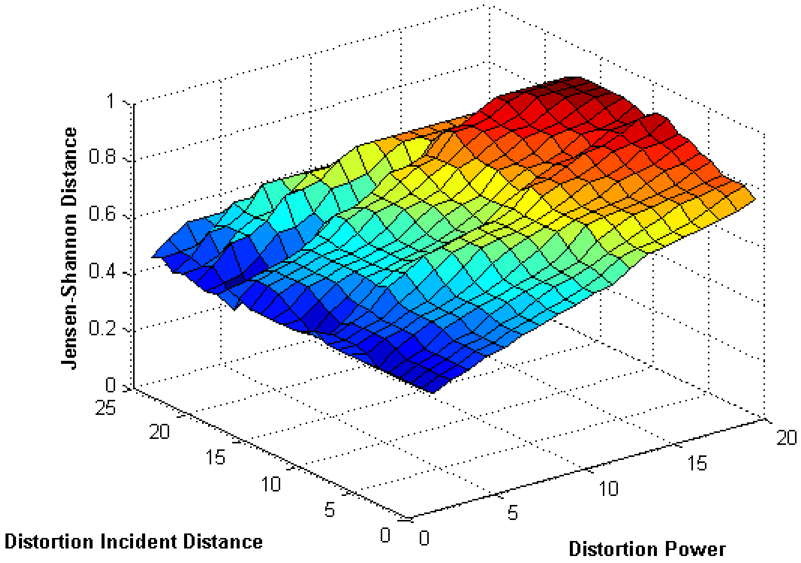


Figure 5.6: Core Path 2 DDLHM JSds: Bias

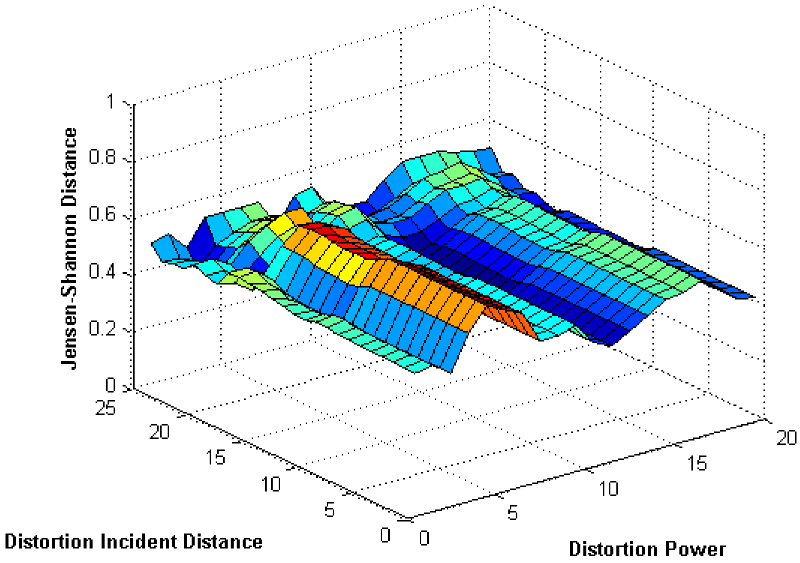


Figure 5.7: Core Path 2 DDLHM JSds: Multipath

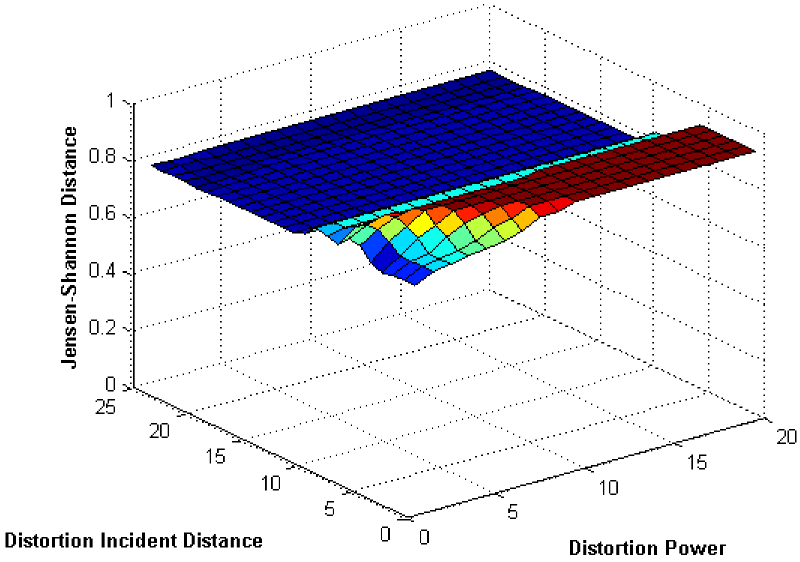


Figure 5.8: Core Path 3 DDLHM JSds: Attenuation

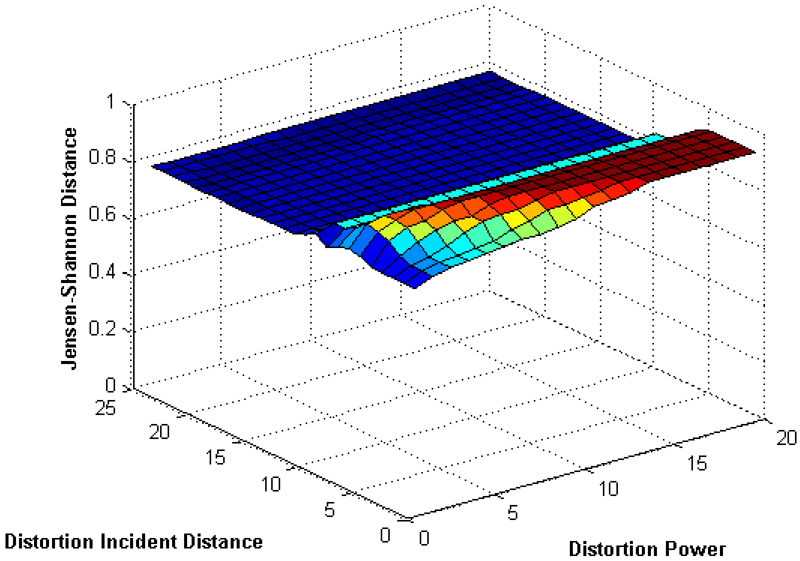


Figure 5.9: Core Path 3 DDLHM JSds: Bias

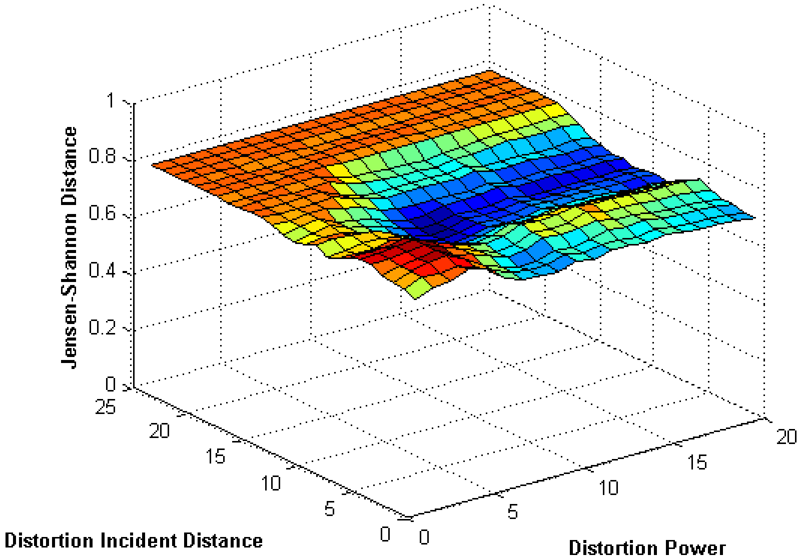


Figure 5.10: Core Path 3 DDLHM JSds: Multipath

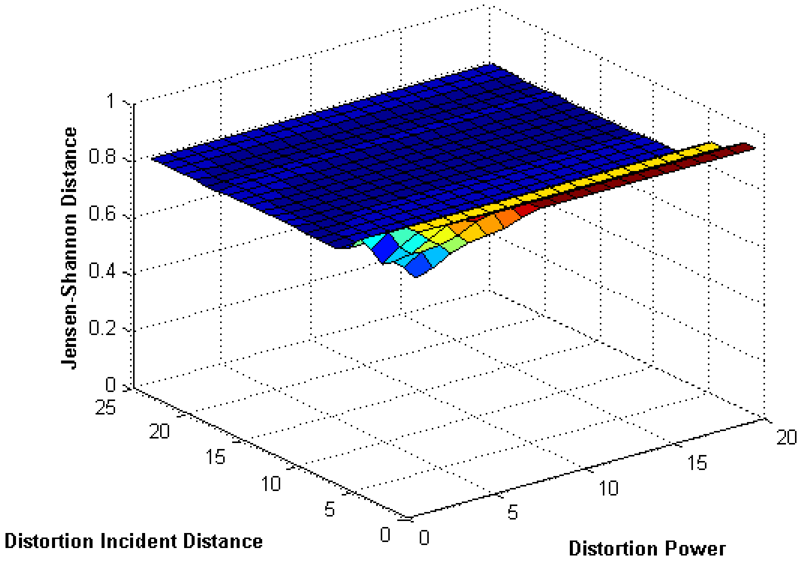


Figure 5.11: Core Path 4 DDLHM JSds: Attenuation

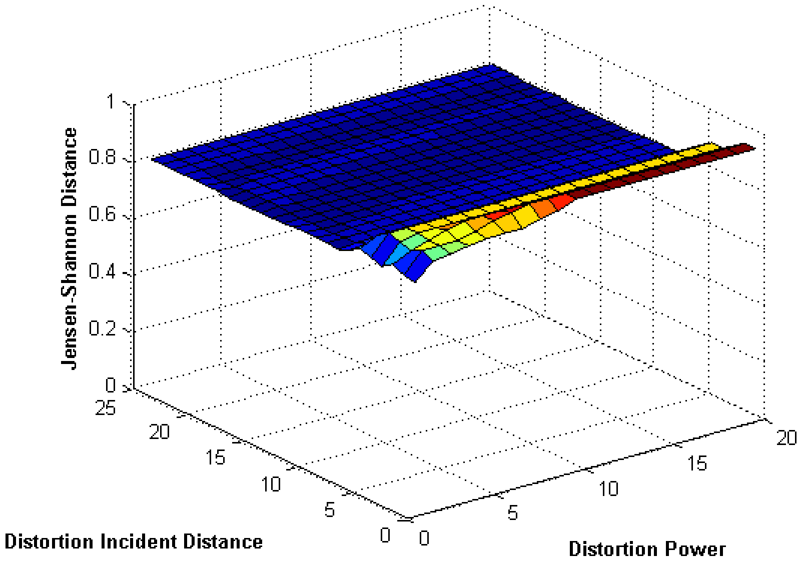


Figure 5.12: Core Path 4 DDLHM JSds: Bias

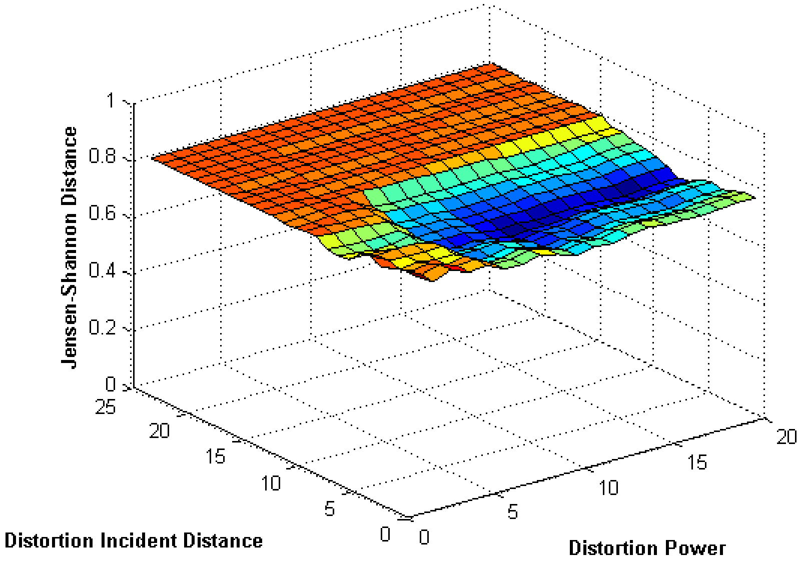


Figure 5.13: Core Path 4 DDLHM JSds: Multipath

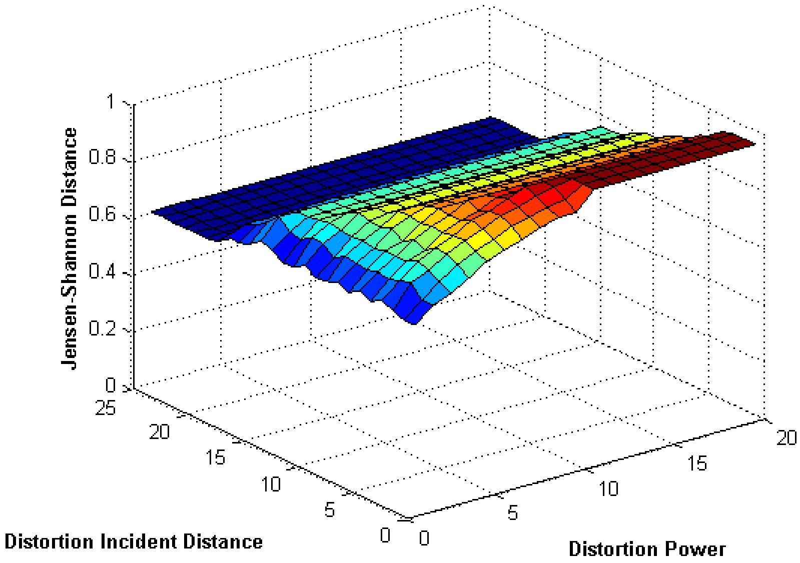


Figure 5.14: Core Path 5 DDLHM JSds: Attenuation

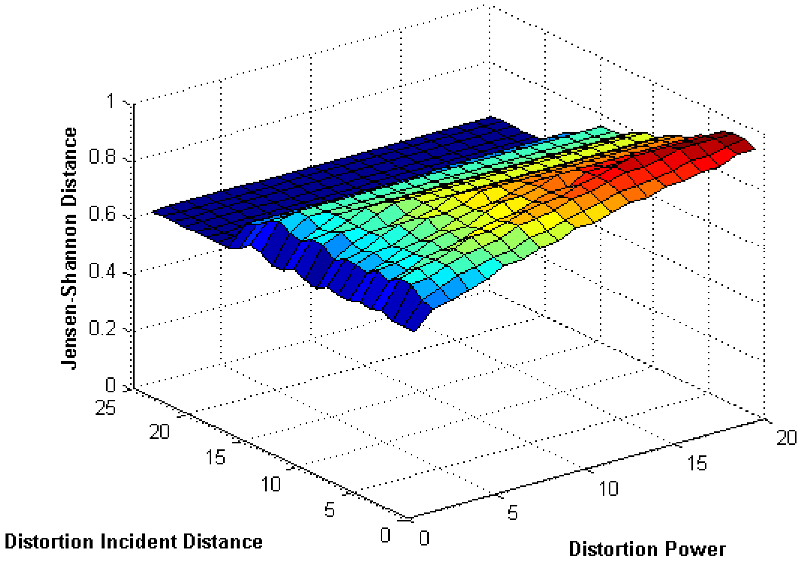


Figure 5.15: Core Path 5 DDLHM JSds: Bias

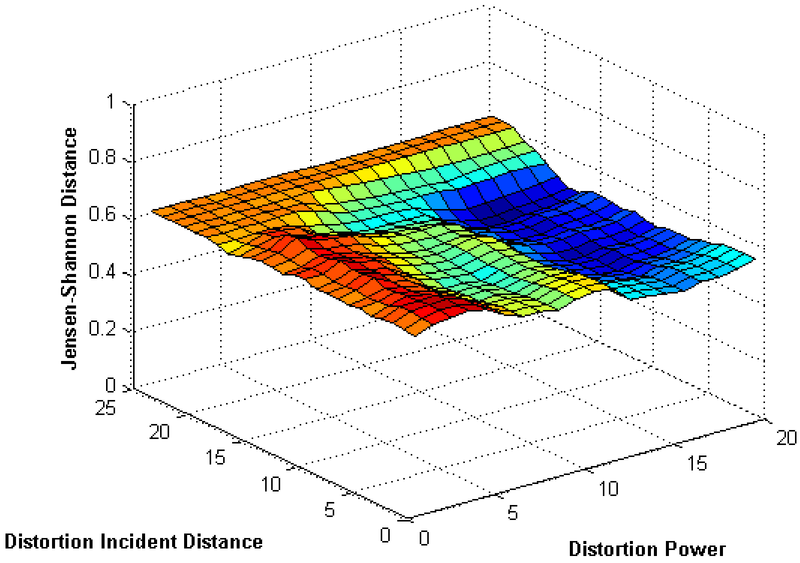


Figure 5.16: Core Path 5 DDLHM JSds: Multipath

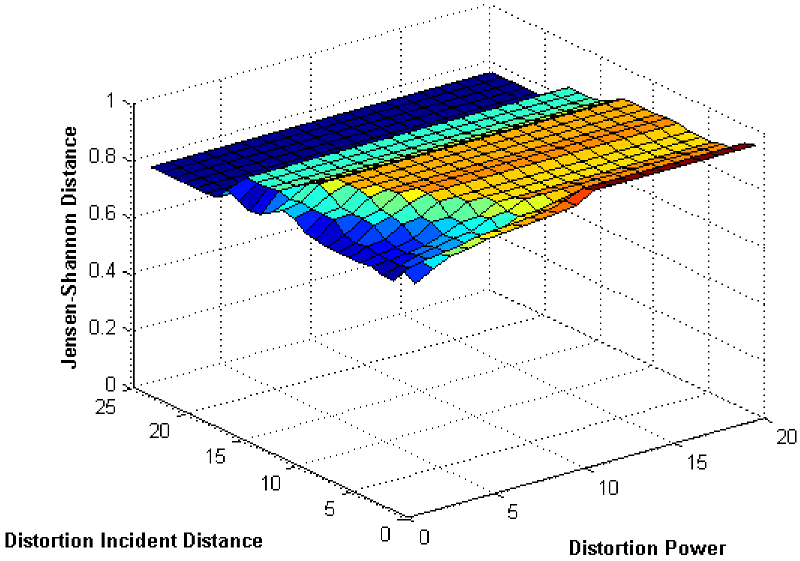


Figure 5.17: Core Path 6 DDLHM JSds: Attenuation

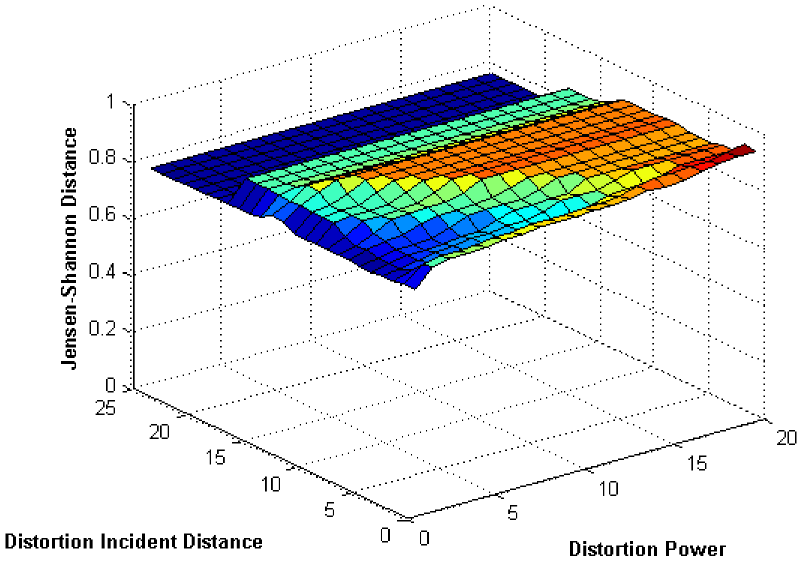


Figure 5.18: Core Path 6 DDLHM JSds: Bias

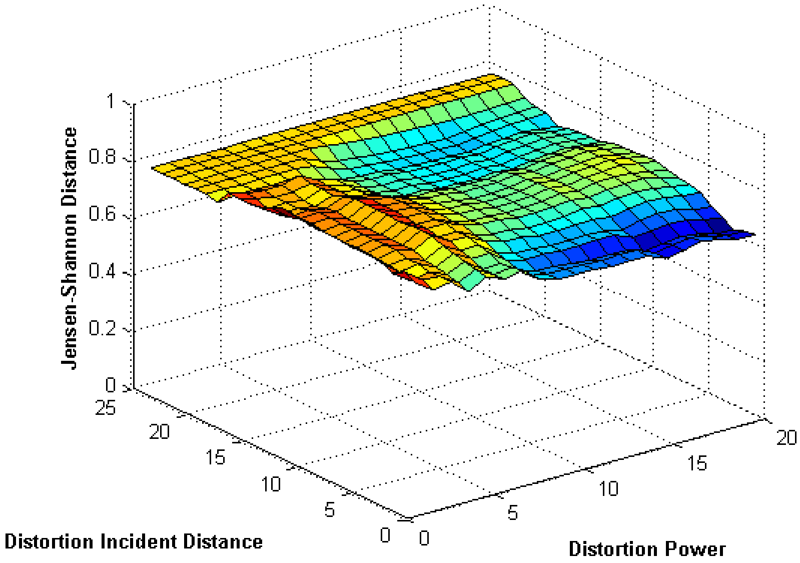


Figure 5.19: Core Path 6 DDLHM JSds: Multipath

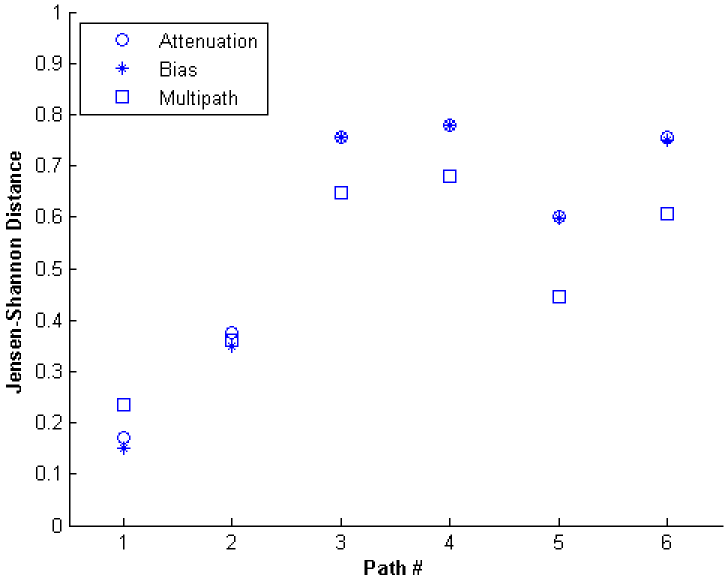


Figure 5.20: Core Best Distortion Matches per Distortion

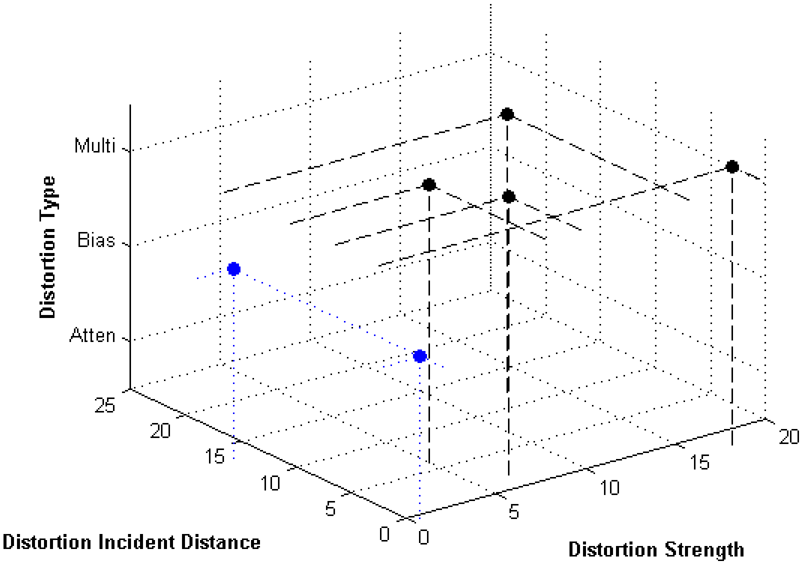


Figure 5.21: Core Best Distortion Matches Overall

For each path of Core, we computed the JSd between the DDLHM of the path and the DDLHM of the reduced set of distortion parameters defined in Section 3.4. We determined: in 3.4.1 that an attenuation-type match of 0.1 JSd is very good, while a match of 0.2 JSd is reasonble, in 3.4.1 that a bias-type match of 0.25 JSd or less is a good match, and in 3.4.1 that a multipath-type match of 0.25 JSd would be a reasonable match, while a match of 0.05 JSd or less would be very good. Keeping in mind the JSd match breakpoints we determined above, we tabulated the minimal JSd per distortion type in Figure 5.20. Path 1 and 2 both have fairly strong indications of bias. Path 1 more so, as it matches below the JSd threshold, path 2 less so as it matches above. Since any signal could be the combination of any number of distortion effects and noise, it is expected that in some cases no distortion profile for any parameterization will match below the 'good match' threshold. In such as case we regard the absolute lowest JSd as the best possible match regardless. Paths 3 through 6 all strongly indicate multipath over bias or attenuation, however all matches are at extremely high JSds, indicating the signal data is highly distorted. From this we can conclude that the Core environment likely has a few fairly low-strength bias distortions, but is dominated by heavy distortions that are most similar to multipath types. Multipath-type distortions can be quite deleterious for localization algorithms as they behave much like lognormal until reaching a signal null, quickly dropping off to extremely low power levels unexpectedly. This type of environment would likely cause fairly few weak errors, but would also cause a similar number of very strong errors.

5.1.3 Distortion Characteristics of the Grid Environment

For each path in the Grid environment, we computed the JSd between the DDLHM of the path's signals and the reduced parameter set. These JSds per distortion type, strength and incident distance are in Figures 5.22 through 5.129. The Grid environment is distinct in that it contains very little other than the Orbit Grid. Unlike the Core and WINLAB environments, the room is mainly empty but for support beams and some tables and equipment against the walls. Since the environment is rarely populated and does not have many structual elements or furniture and is floored entirely in tile, it should have fairly distinct propagation behavior. Since each of the 400 nodes in the Orbit Grid has two wireless cards on it, it is possible to have any node record signal strengths while any other node transmits. Due to its unique structure, we recorded signal vectors over 18 of the 20 vertical vectors and 18 of the 20 horizontal vectors that had all networking cards operating. As can be seen in Figure 5.130, the vast majority of best-matching distortions were of the multipath type. In many cases, the multipath matches are much more strongly indicated than the attenuation and bias types, even when the JSd value is not below that required for a good match. This can especially be seen for paths 15 through 18, where the absolute minimum JSd value for both attenuation and bias distortion types are the same and 0.1 to 0.2 JSd higher than multipath's. Only two paths, 36 and 23, match any distortion type other than multipath. Even so, path 25's match is fairly indecisive. The best JSd matches float between 0.2 and just above 0.4, with an average just above 0.3 JSd. Given the fact that nearly all distortions match quite distinctly as multipath, we expect the Grid environment would actually present a significant problem to most localization algorithms. Since multipath propagation causes a deep fade at points where the lognormal model would predict a much stronger signal, it would be reasonable to presume that model-based localization algorithms would experience significant difficulty localizing as one of their fundamental operating assumptions would occasionally be quite incorrect.

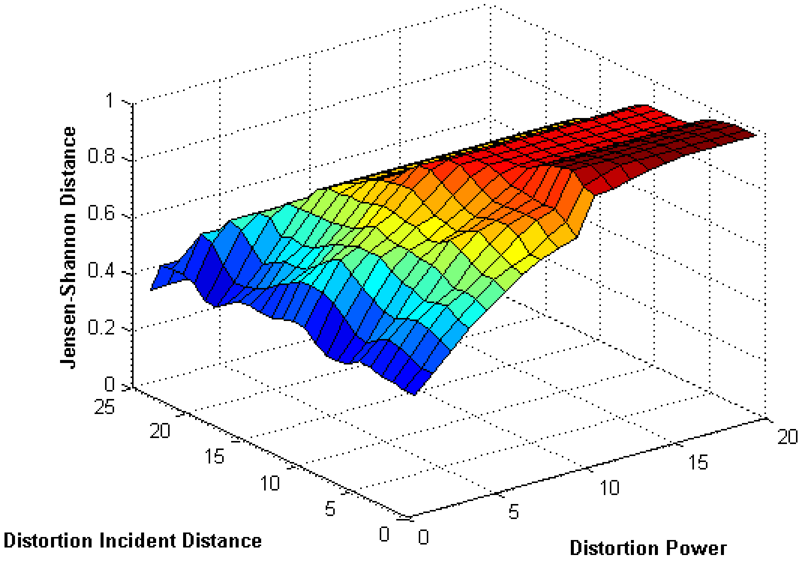


Figure 5.22: Path 1 Attenuation DDLHM JSds

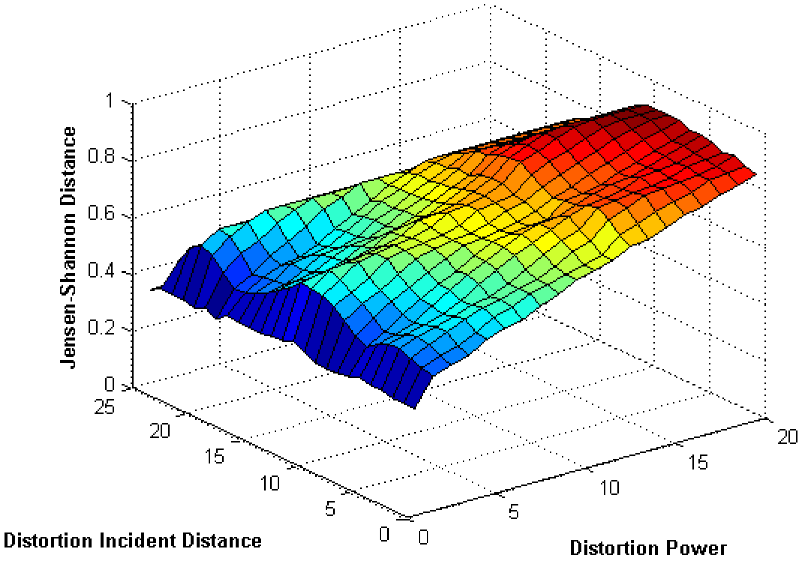


Figure 5.23: Path 1 Bias DDLHM JSds

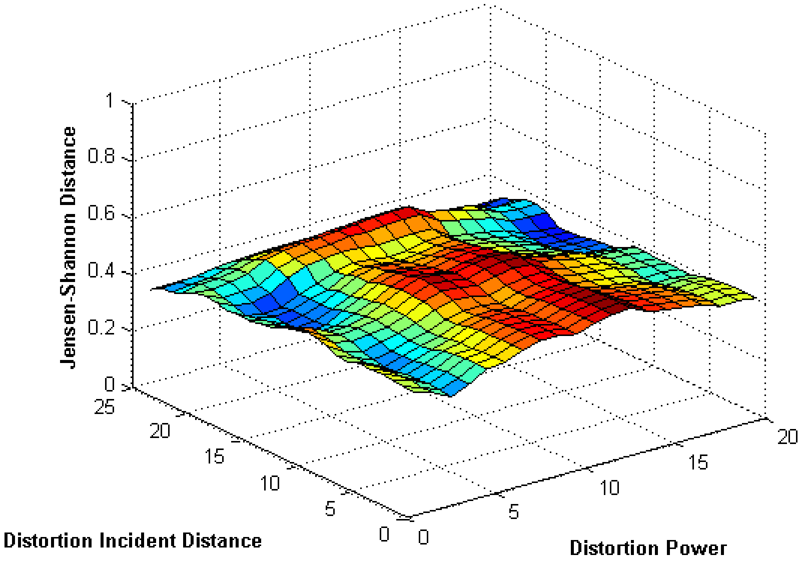


Figure 5.24: Path 1 Multipath DDLHM JSds

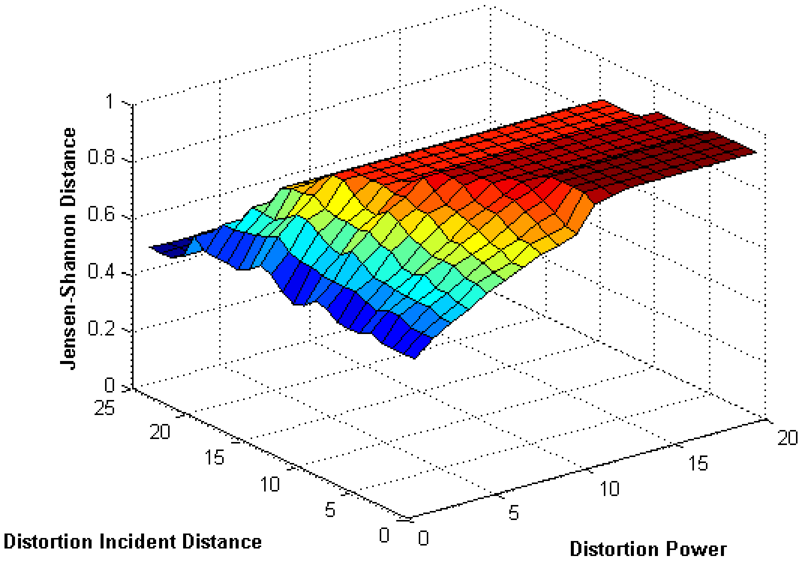


Figure 5.25: Path 2 Attenuation DDLHM JSds

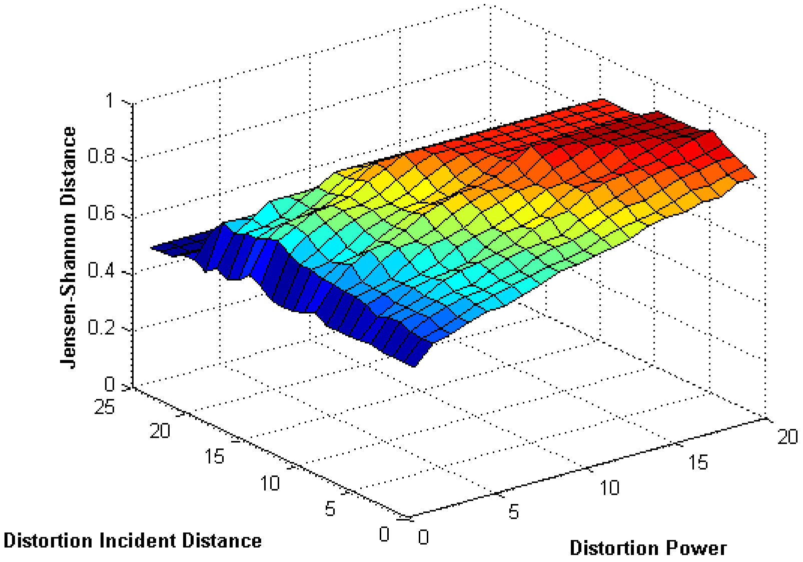


Figure 5.26: Path 2 Bias DDLHM JSds

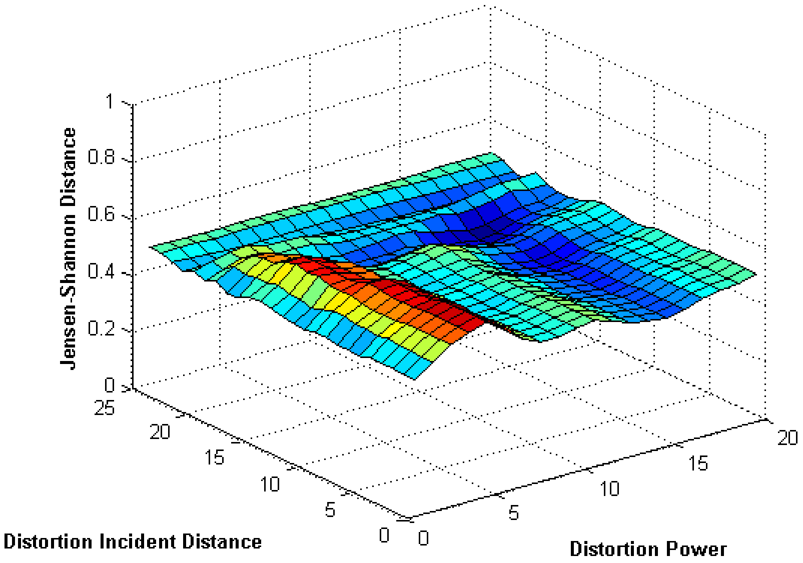


Figure 5.27: Path 2 Multipath DDLHM JSds

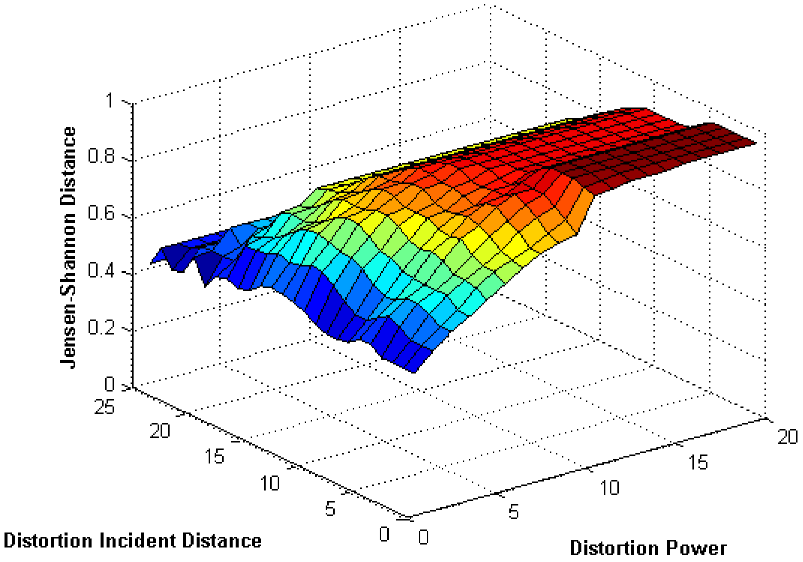


Figure 5.28: Path 3 Attenuation DDLHM JSds

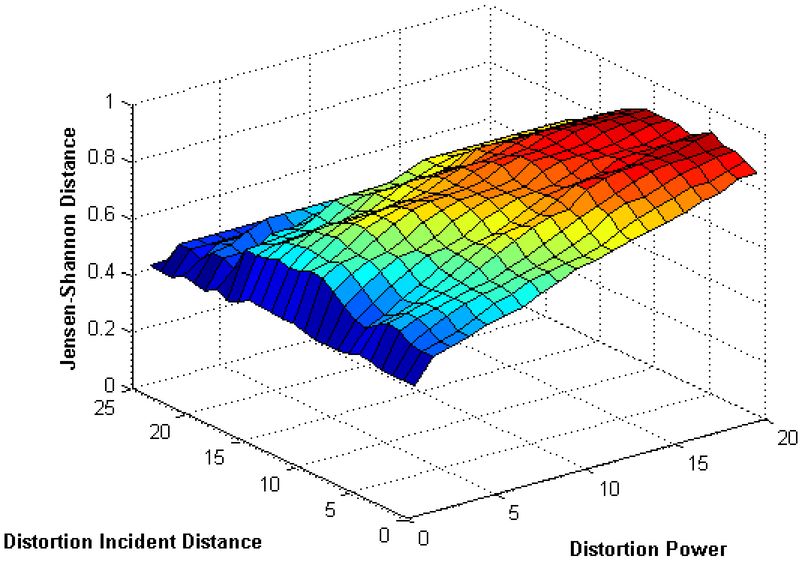


Figure 5.29: Path 3 Bias DDLHM JSds

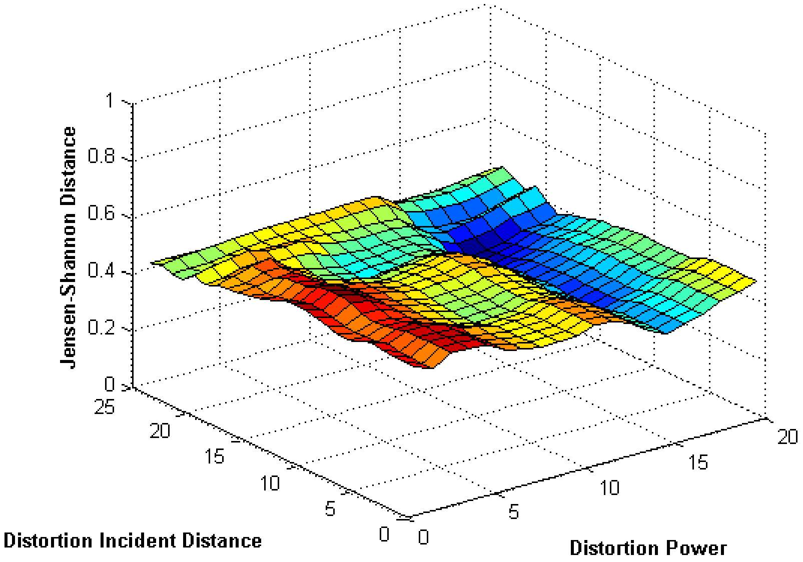


Figure 5.30: Path 3 Multipath DDLHM JSds

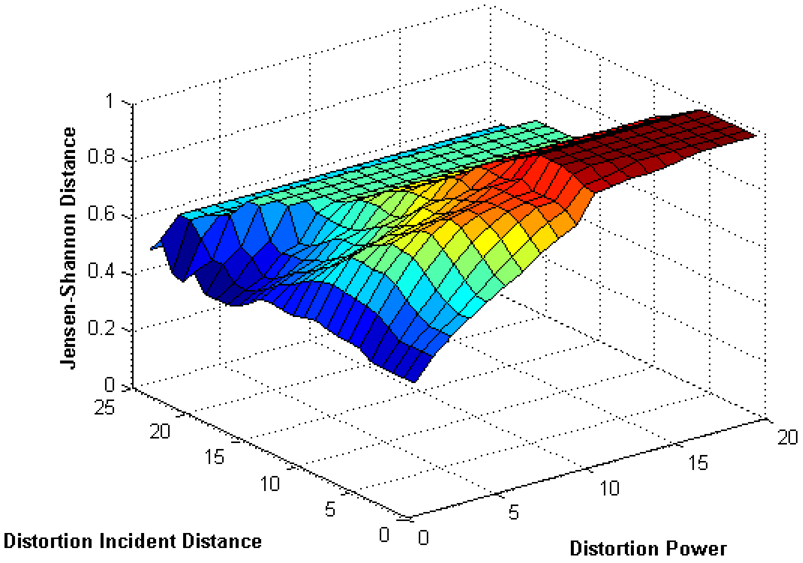


Figure 5.31: Path 4 Attenuation DDLHM JSds

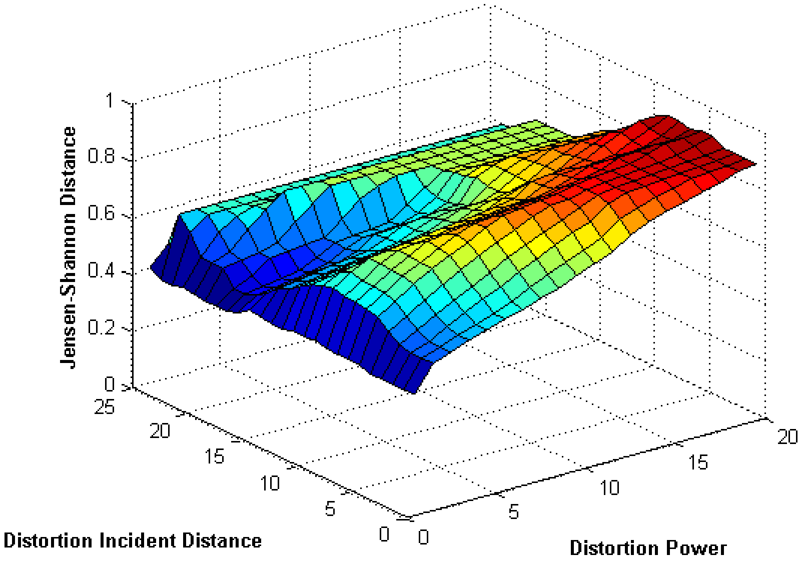


Figure 5.32: Path 4 Bias DDLHM JSds

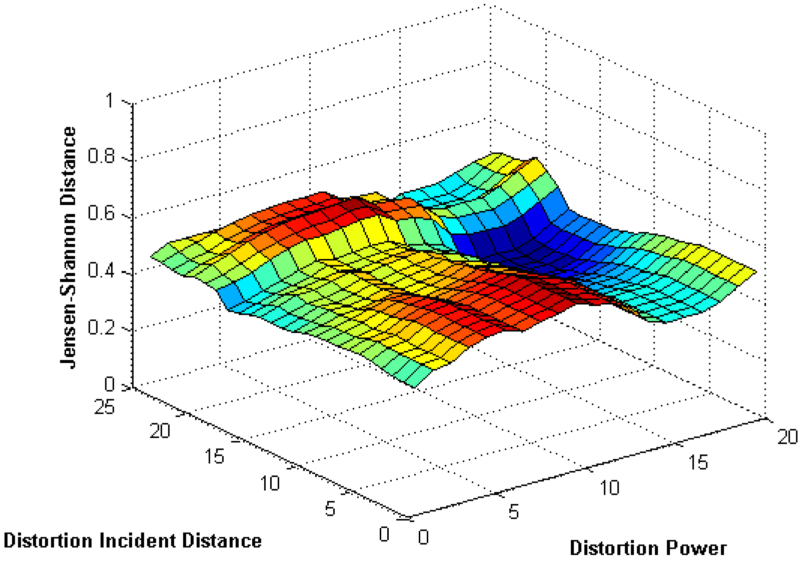


Figure 5.33: Path 4 Multipath DDLHM JSds

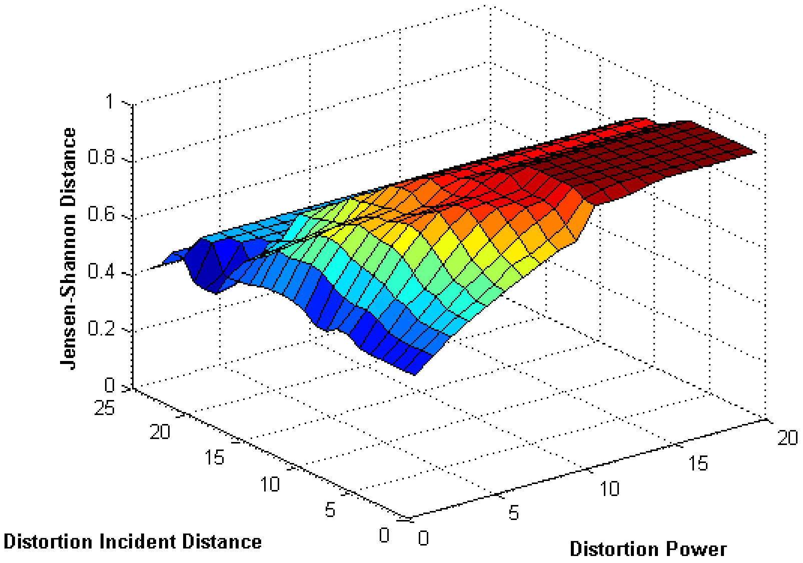


Figure 5.34: Path 5 Attenuation DDLHM JSds

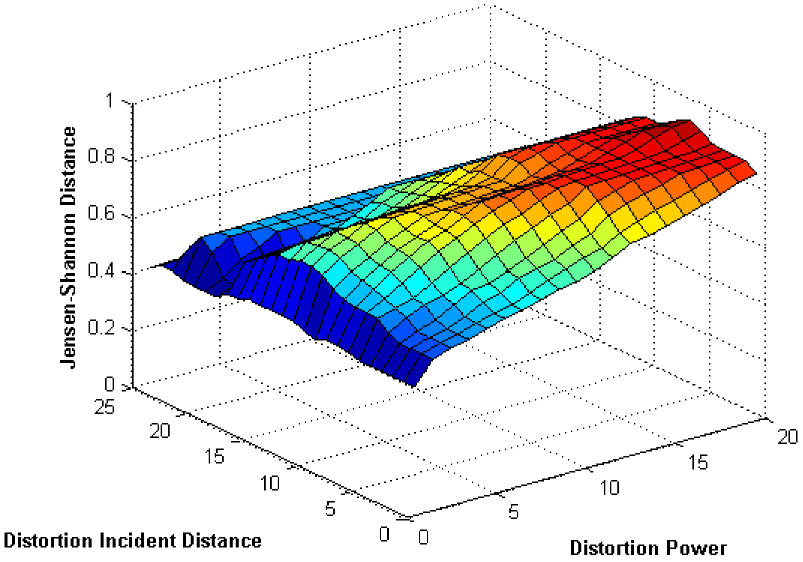


Figure 5.35: Path 5 Bias DDLHM JSds

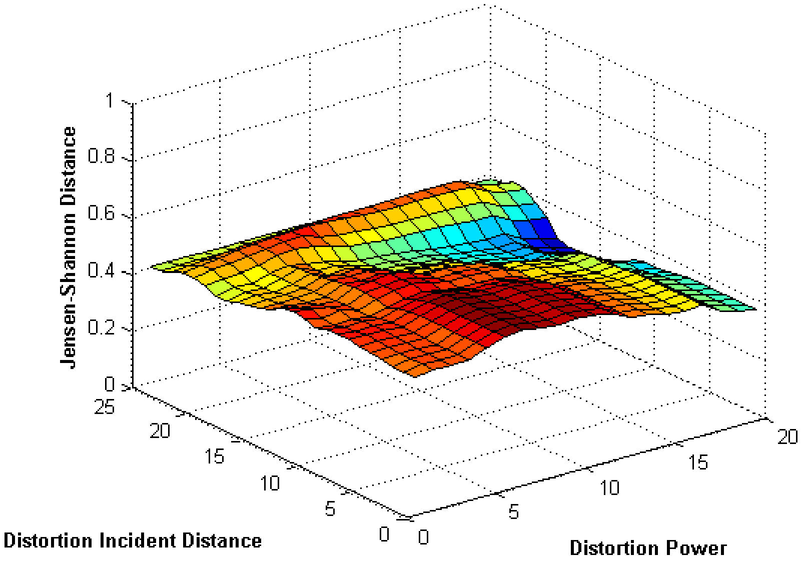


Figure 5.36: Path 5 Multipath DDLHM JSds

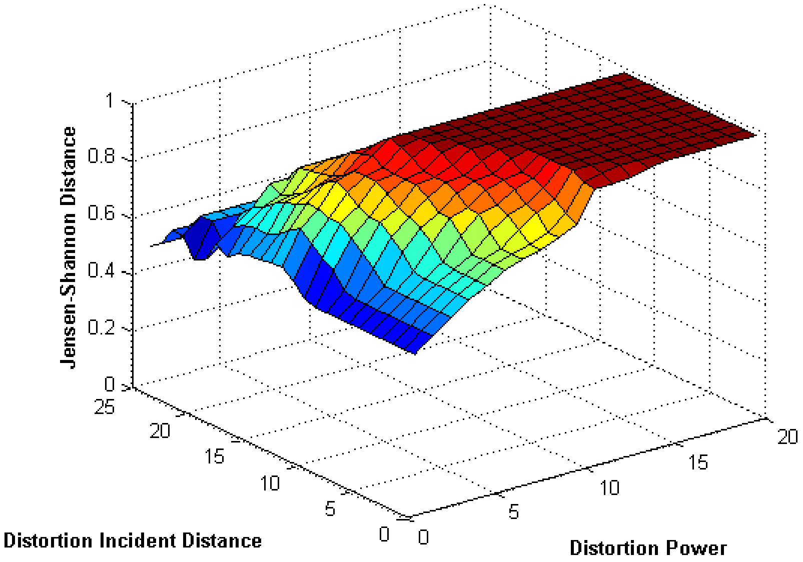


Figure 5.37: Path 6 Attenuation DDLHM JSds

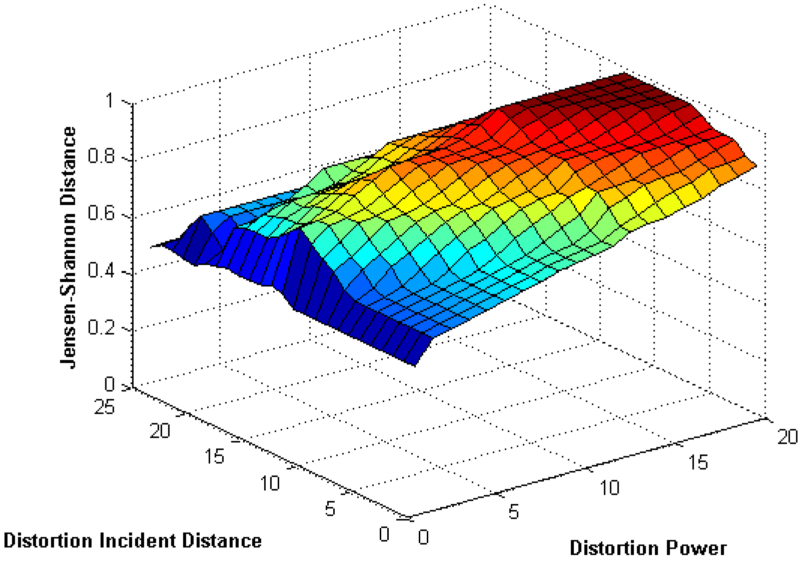


Figure 5.38: Path 6 Bias DDLHM JSds

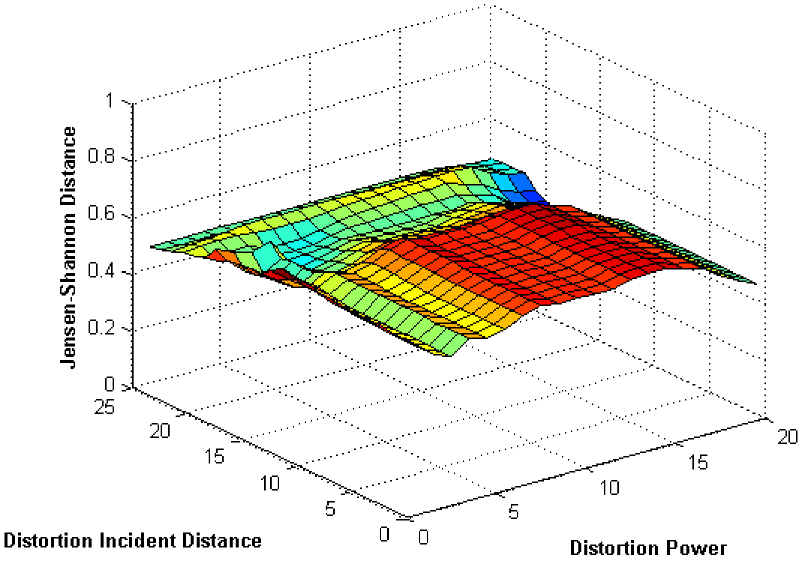


Figure 5.39: Path 6 Multipath DDLHM JSds

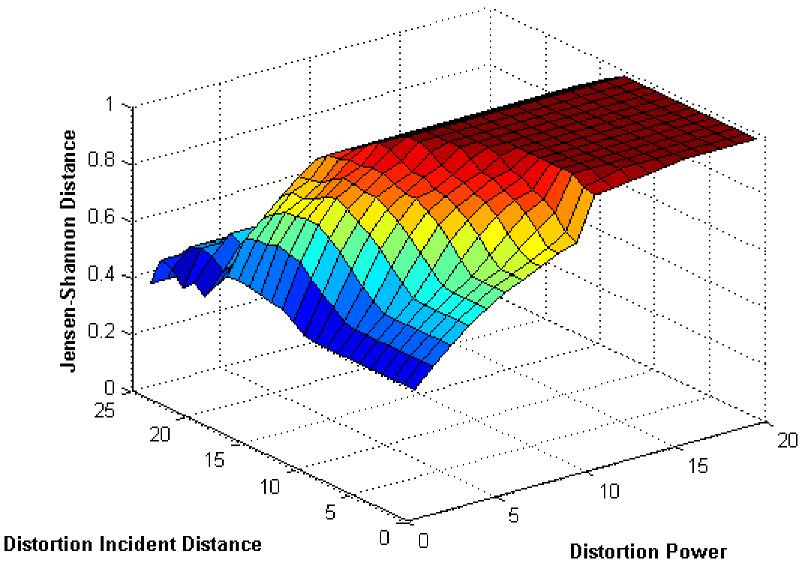


Figure 5.40: Path 7 Attenuation DDLHM JSds

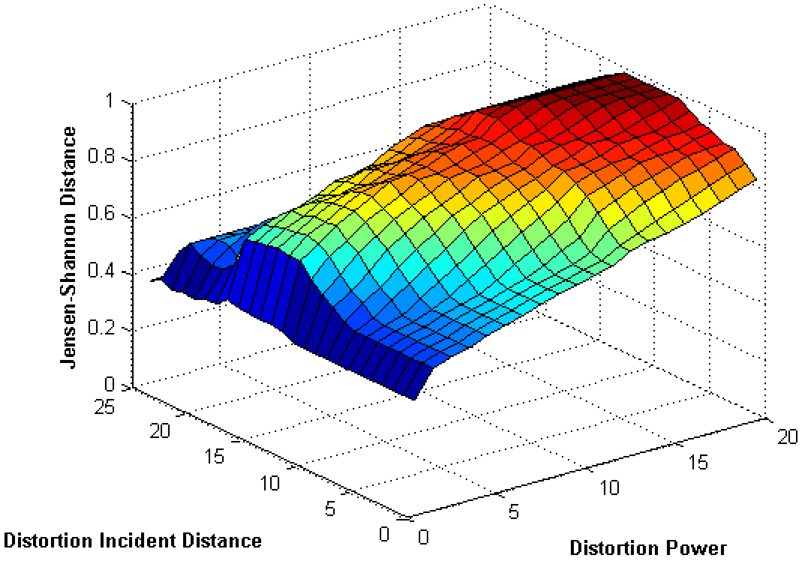


Figure 5.41: Path 7 Bias DDLHM JSds

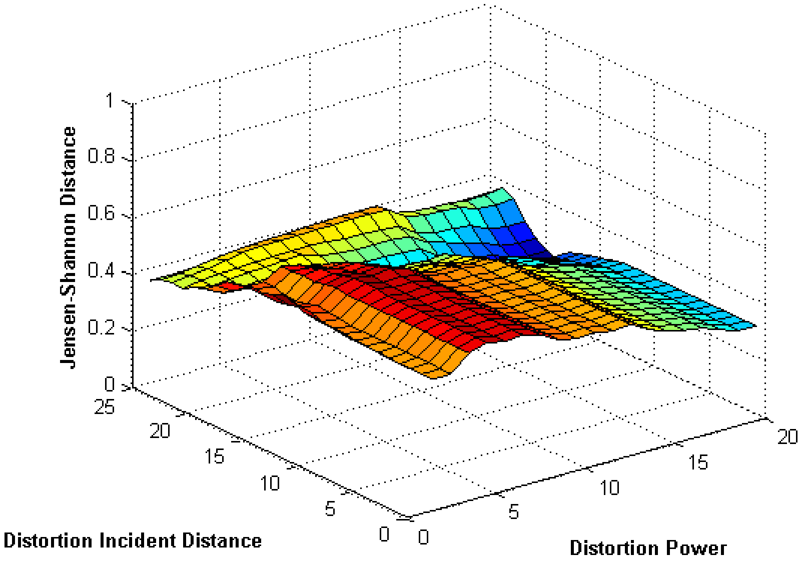


Figure 5.42: Path 7 Multipath DDLHM JSds

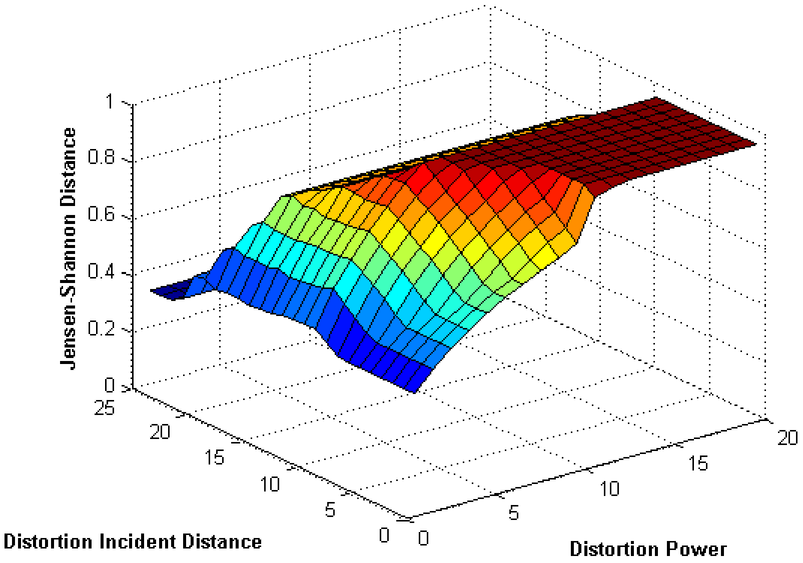


Figure 5.43: Path 8 Attenuation DDLHM JSds

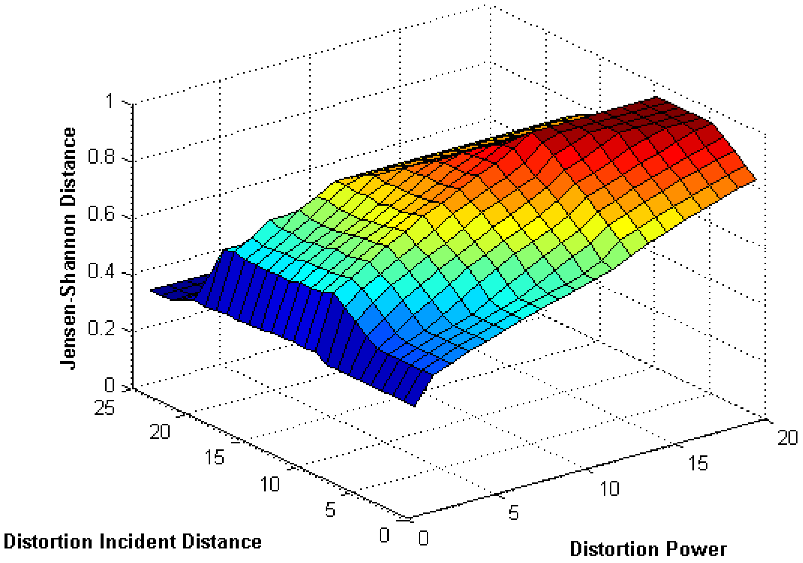


Figure 5.44: Path 8 Bias DDLHM JSds

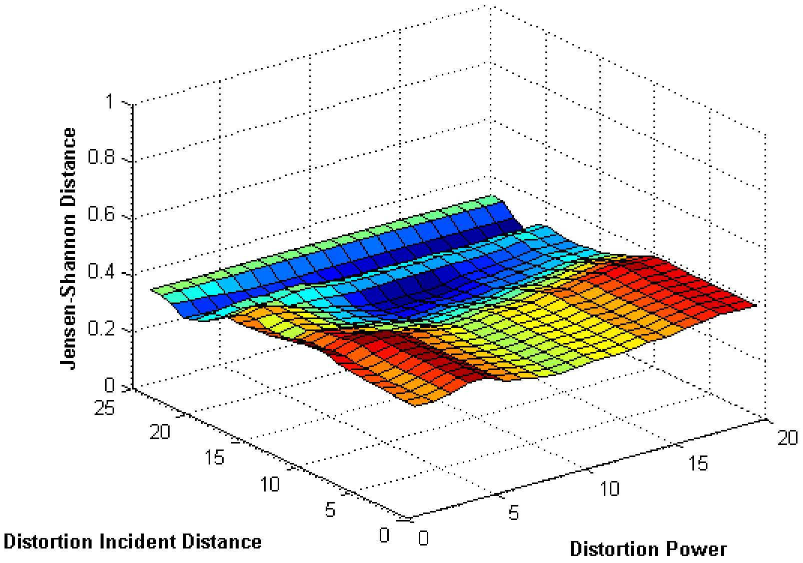


Figure 5.45: Path 8 Multipath DDLHM JSds

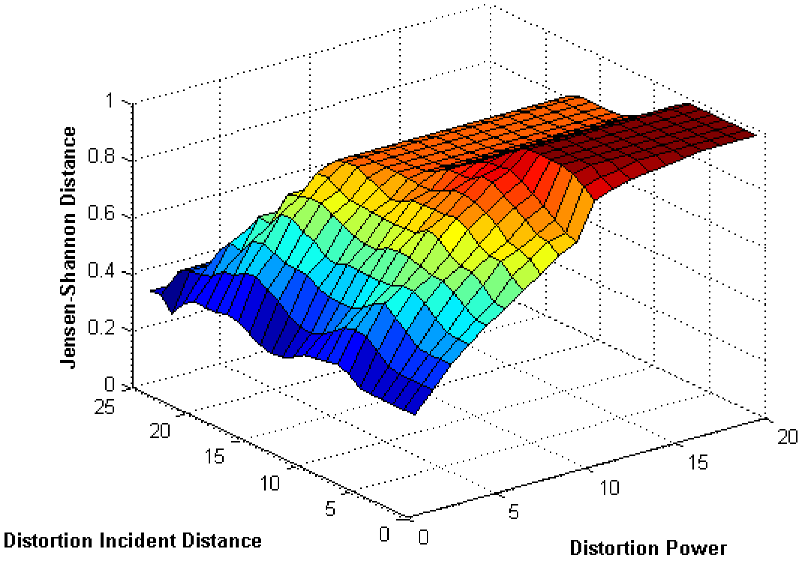


Figure 5.46: Path 9 Attenuation DDLHM JSds

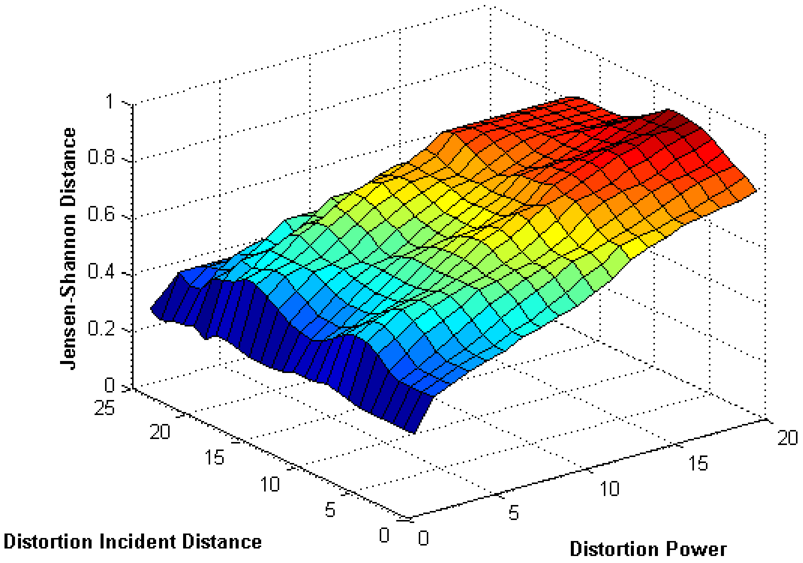


Figure 5.47: Path 9 Bias DDLHM JSds

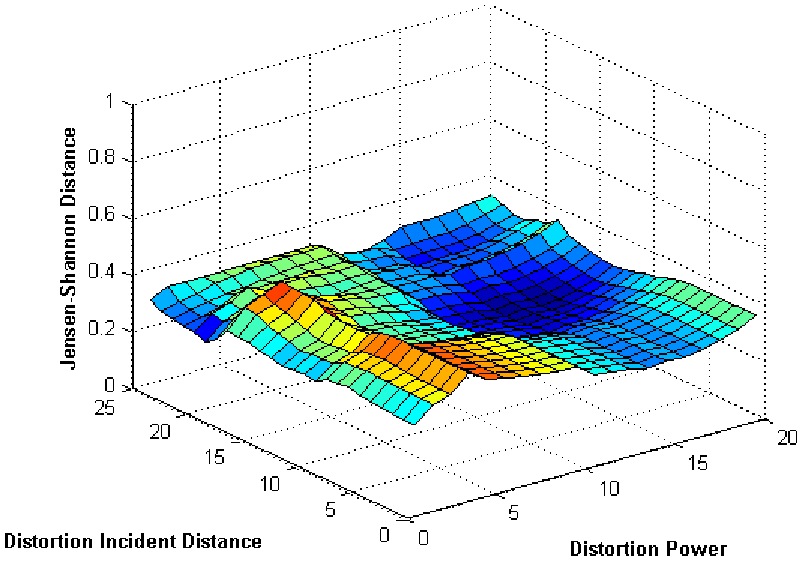


Figure 5.48: Path 9 Multipath DDLHM JSds

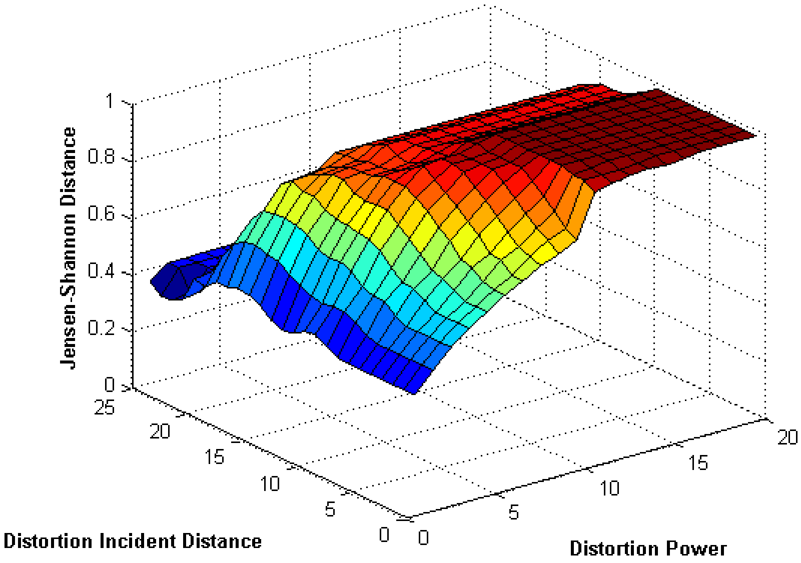


Figure 5.49: Path 10 Attenuation DDLHM JSds

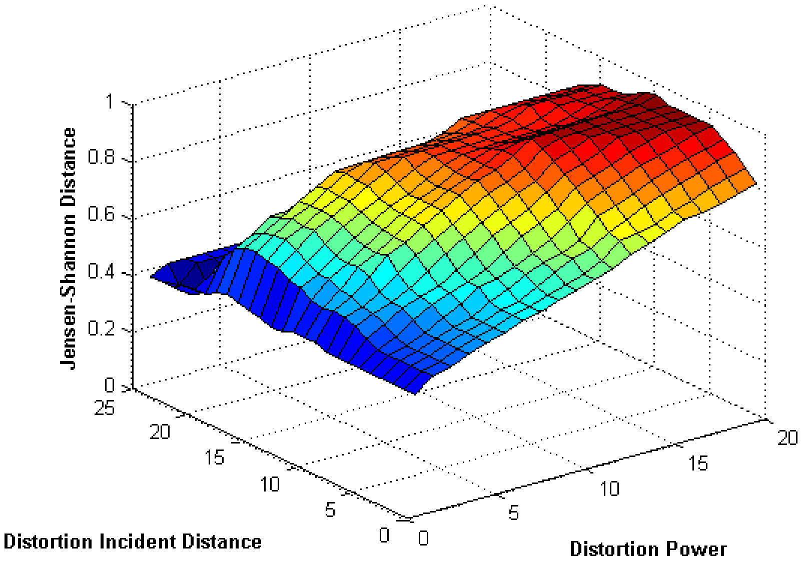


Figure 5.50: Path 10 Bias DDLHM JSds

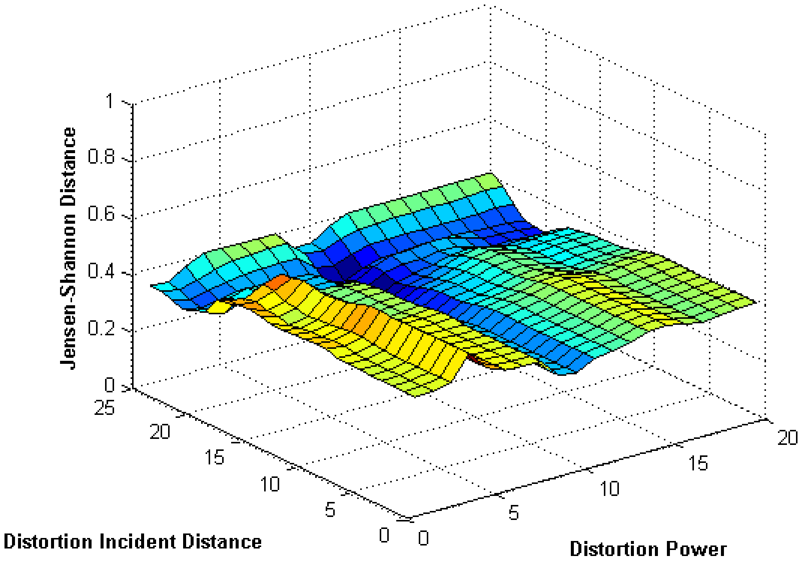


Figure 5.51: Path 10 Multipath DDLHM JSds

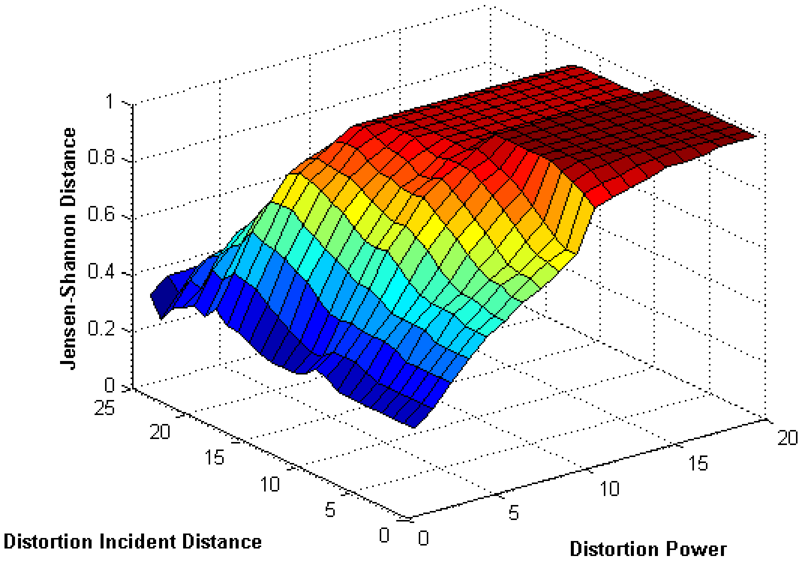


Figure 5.52: Path 11 Attenuation DDLHM JSds

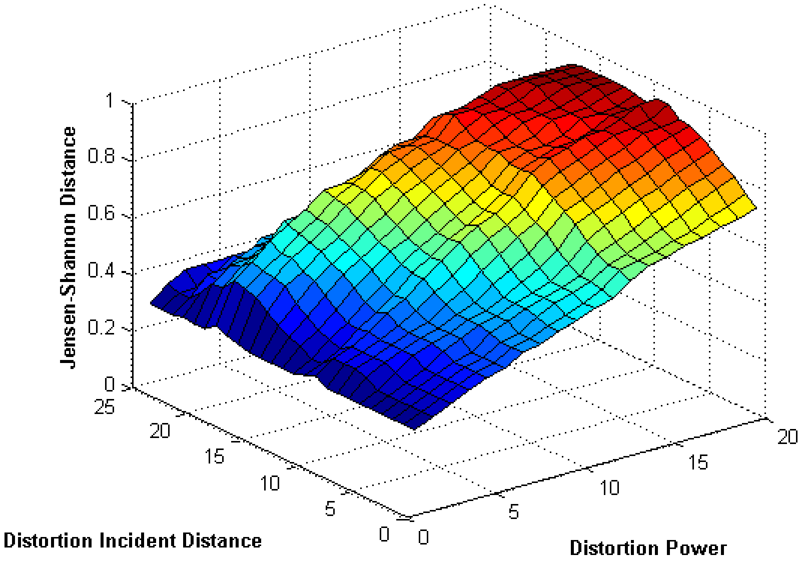


Figure 5.53: Path 11 Bias DDLHM JSds

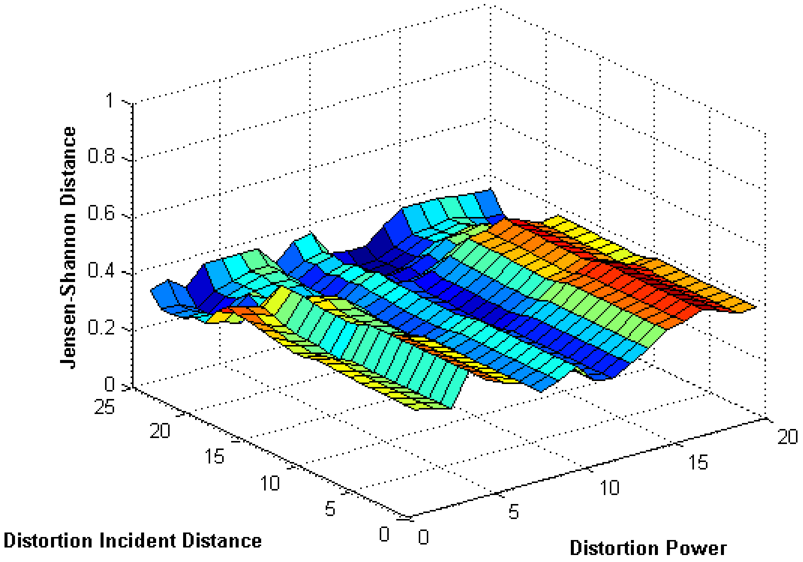


Figure 5.54: Path 11 Multipath DDLHM JSds

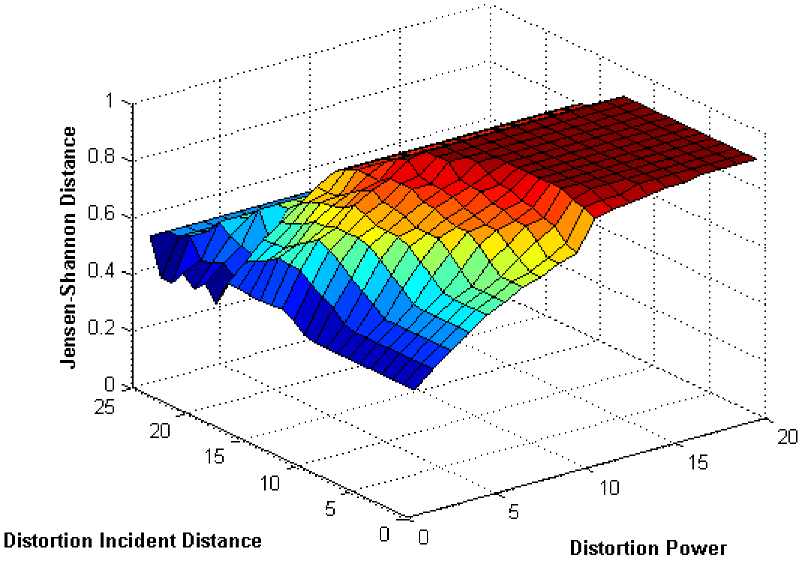


Figure 5.55: Path 12 Attenuation DDLHM JSds

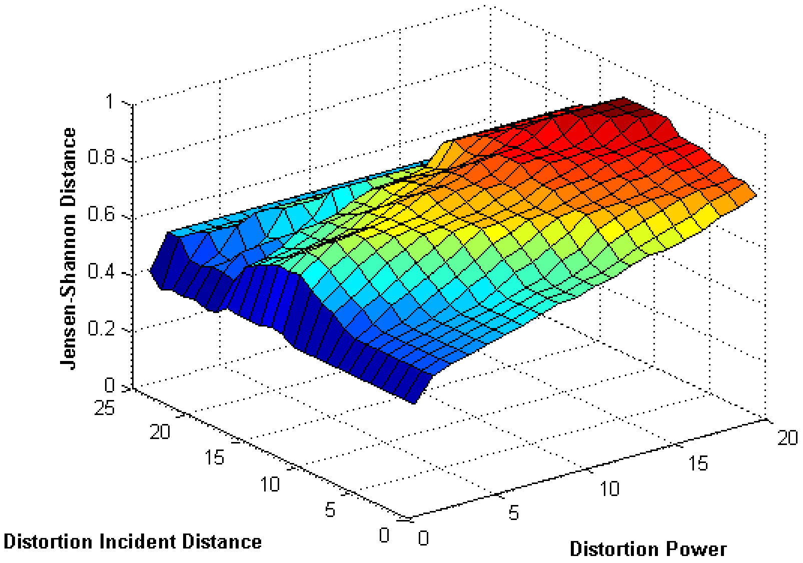


Figure 5.56: Path 12 Bias DDLHM JSds

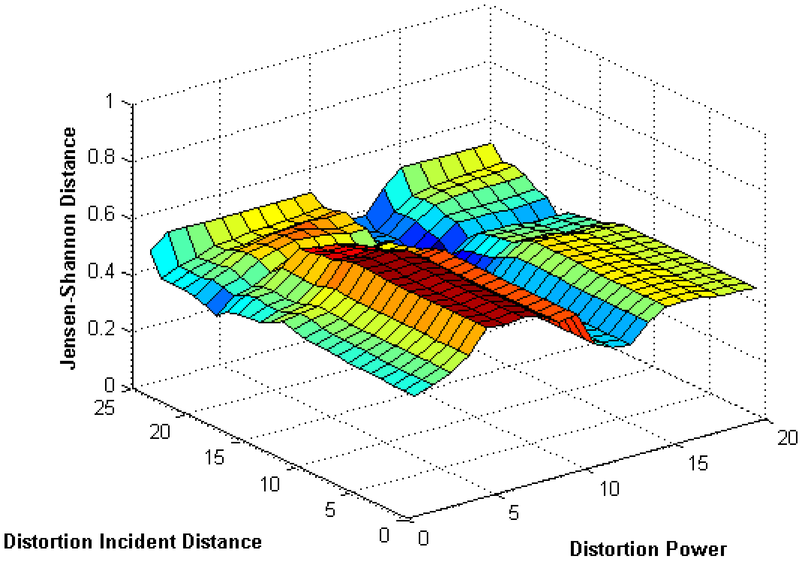


Figure 5.57: Path 12 Multipath DDLHM JSds

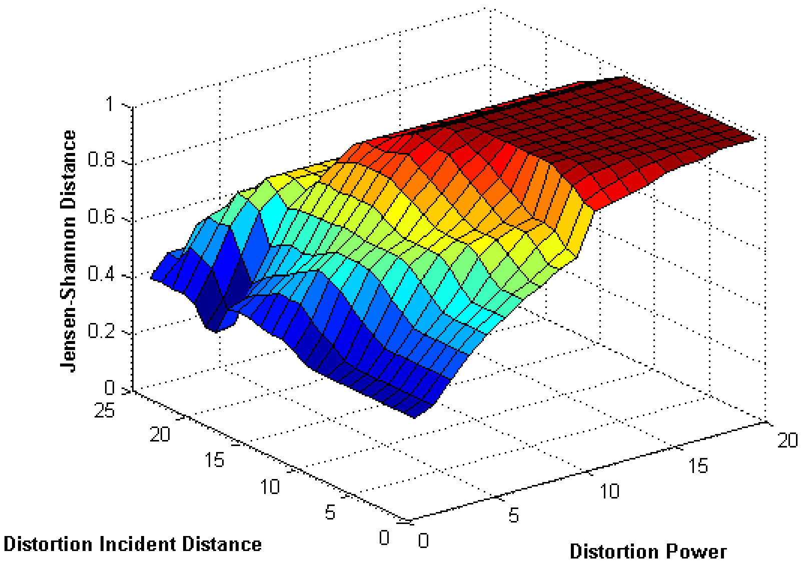


Figure 5.58: Path 13 Attenuation DDLHM JSds

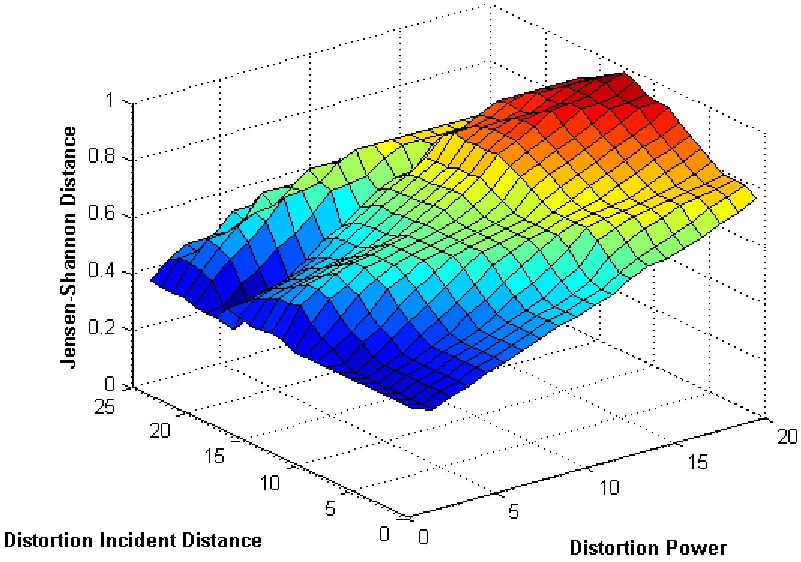


Figure 5.59: Path 13 Bias DDLHM JSds

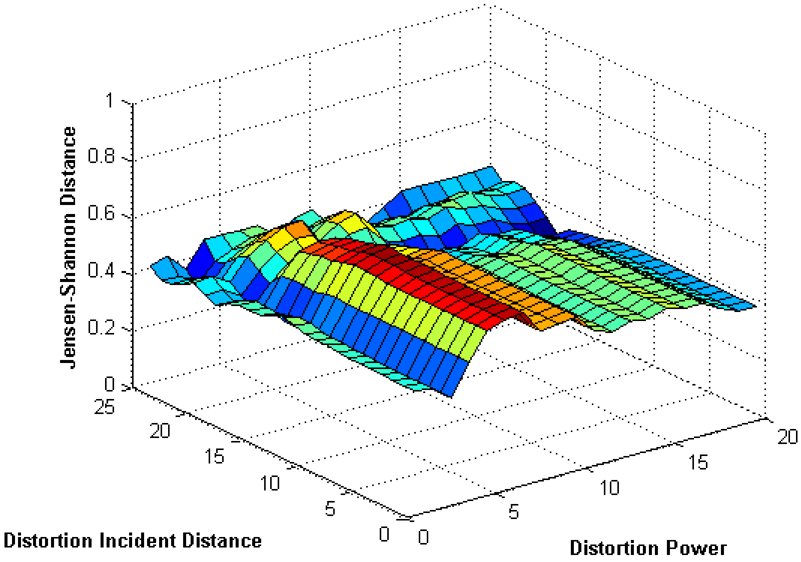


Figure 5.60: Path 13 Multipath DDLHM JSds

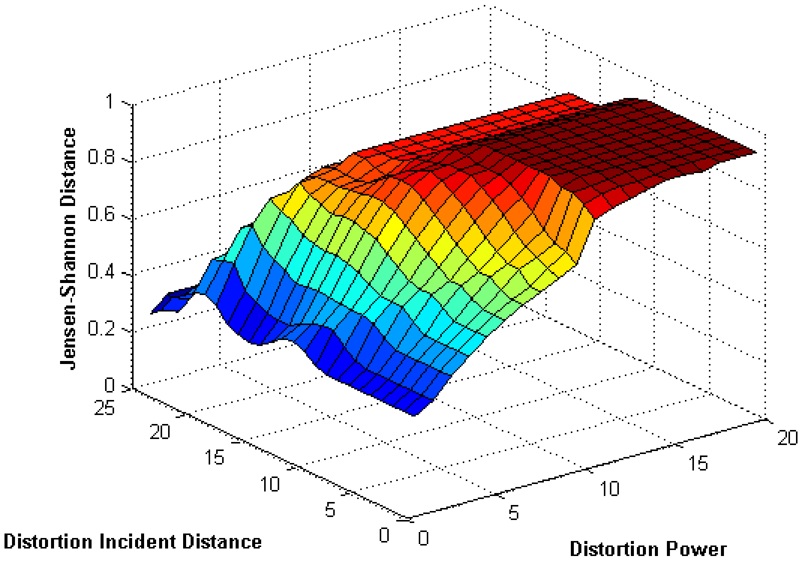


Figure 5.61: Path 14 Attenuation DDLHM JSds

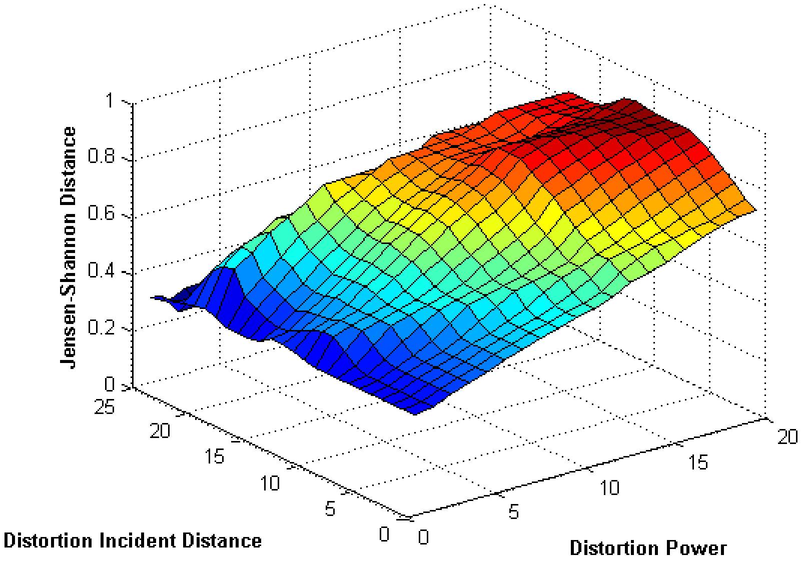


Figure 5.62: Path 14 Bias DDLHM JSds

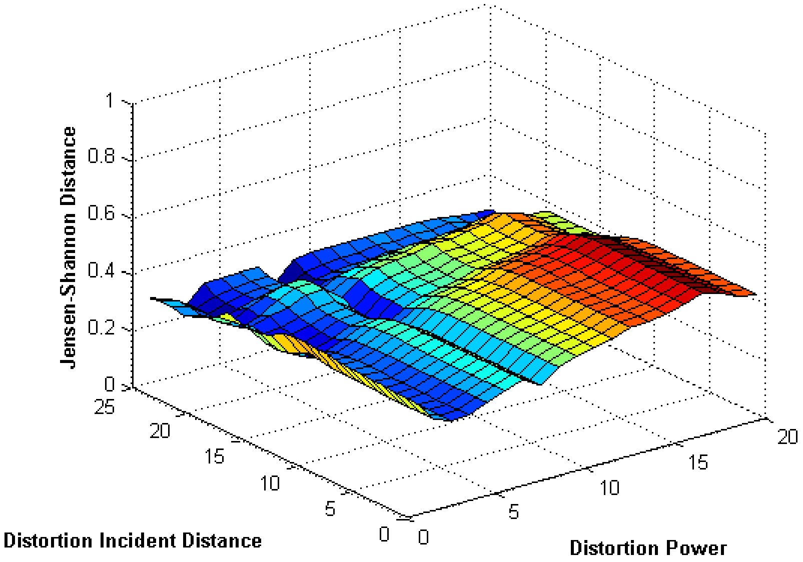


Figure 5.63: Path 14 Multipath DDLHM JSds

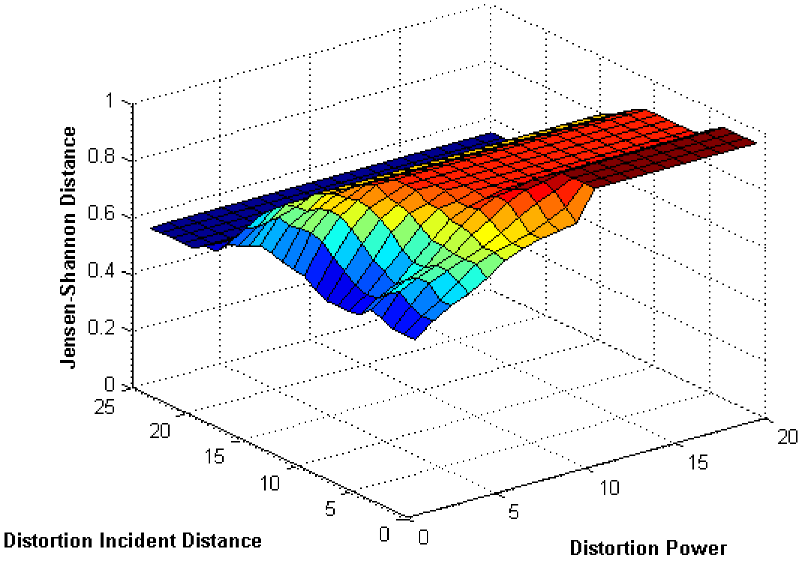


Figure 5.64: Path 15 Attenuation DDLHM JSds

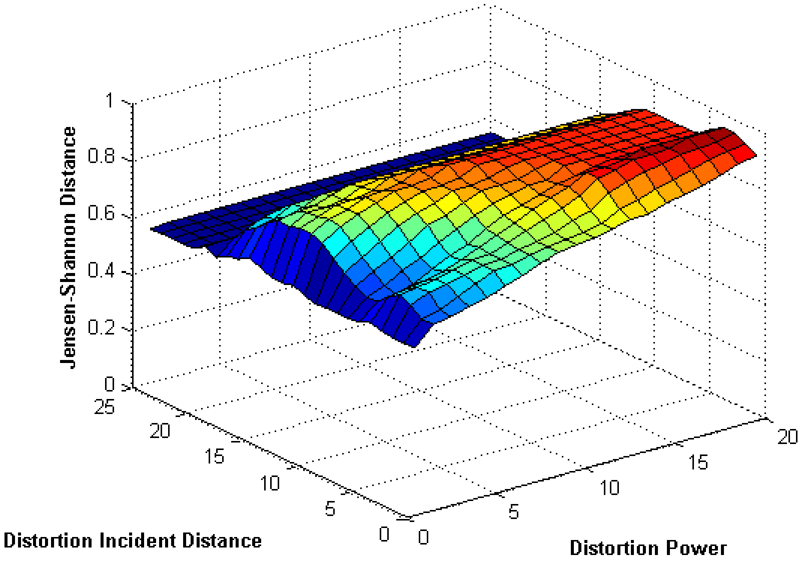


Figure 5.65: Path 15 Bias DDLHM JSds

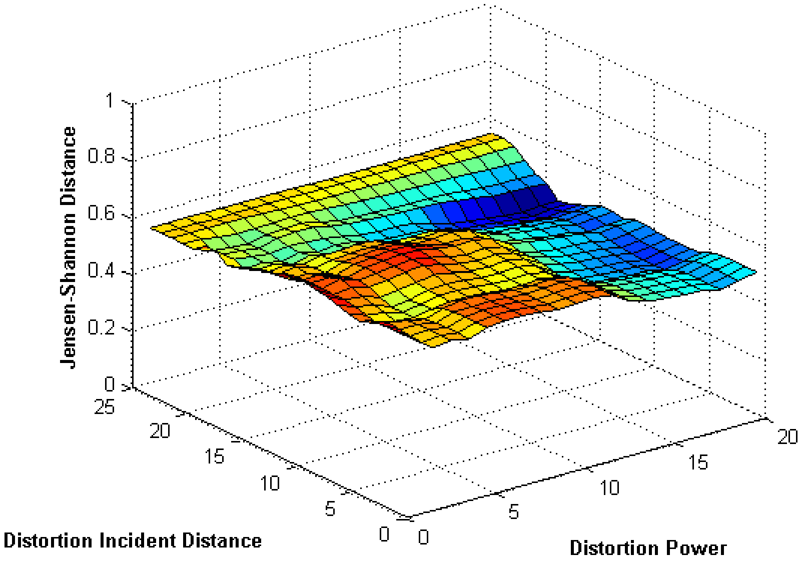


Figure 5.66: Path 15 Multipath DDLHM JSds

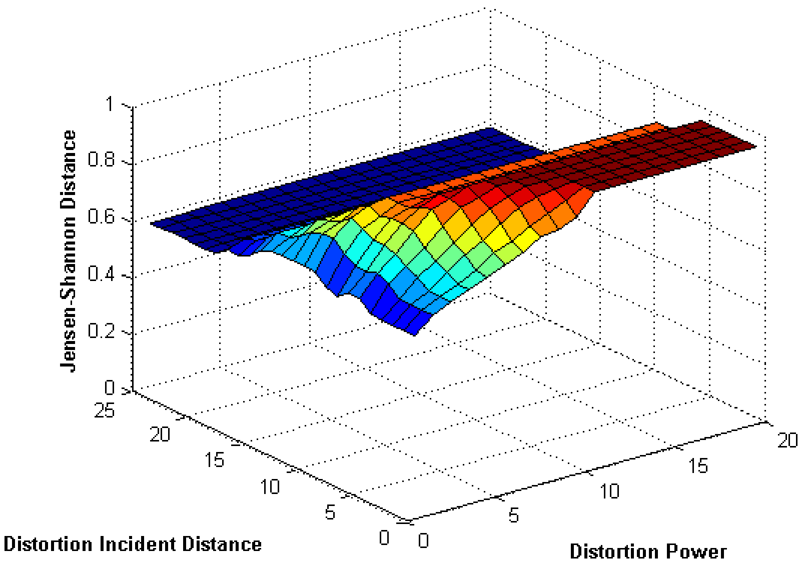


Figure 5.67: Path 16 Attenuation DDLHM JSds

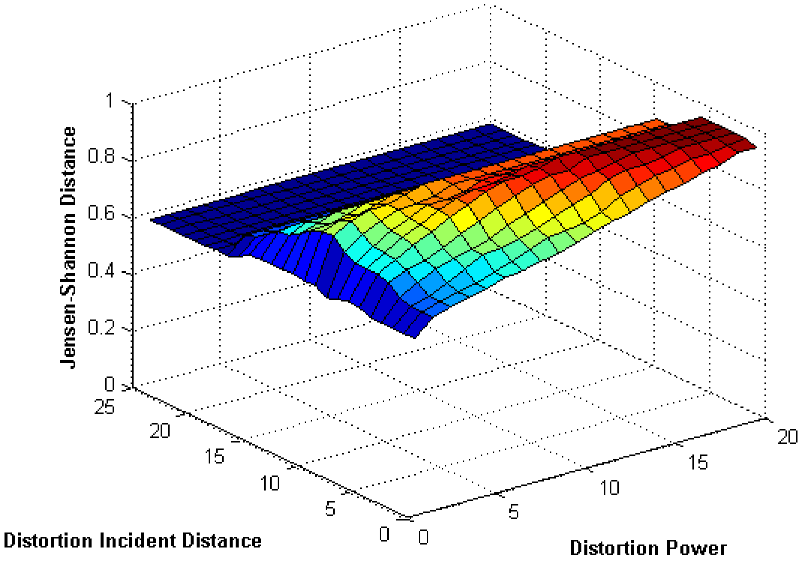


Figure 5.68: Path 16 Bias DDLHM JSds

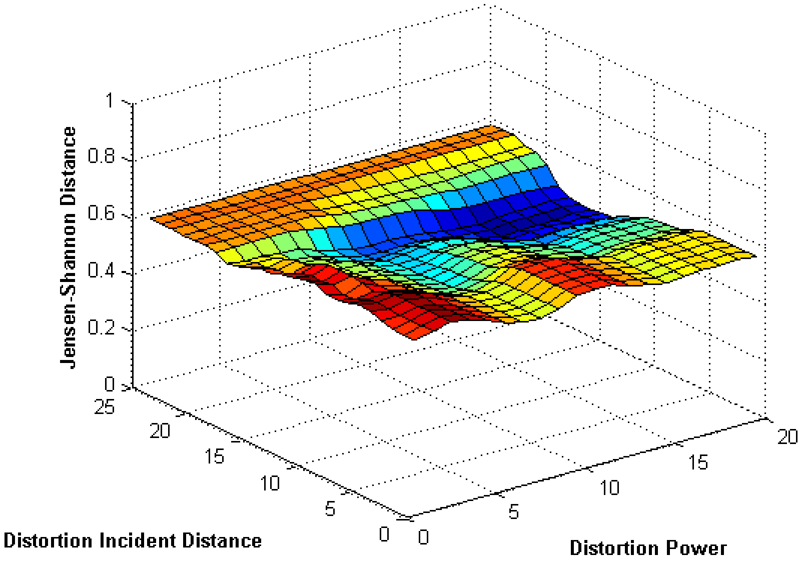


Figure 5.69: Path 16 Multipath DDLHM JSds

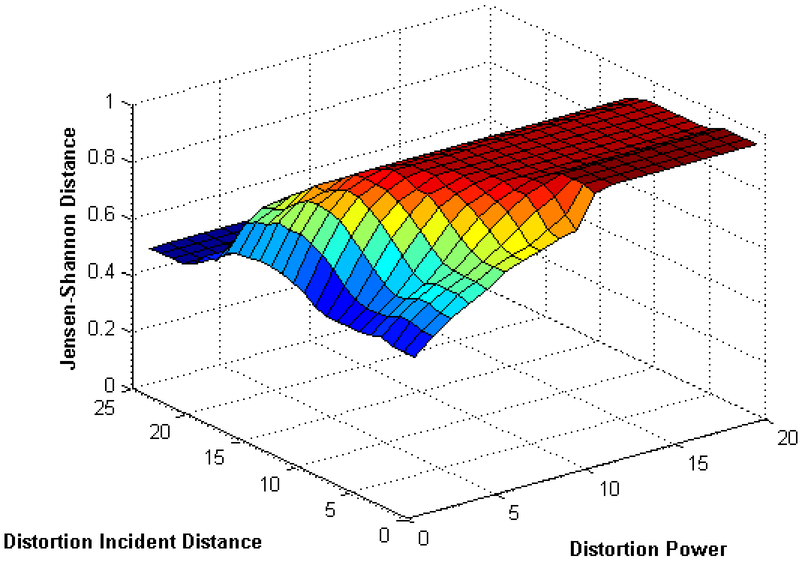


Figure 5.70: Path 17 Attenuation DDLHM JSds

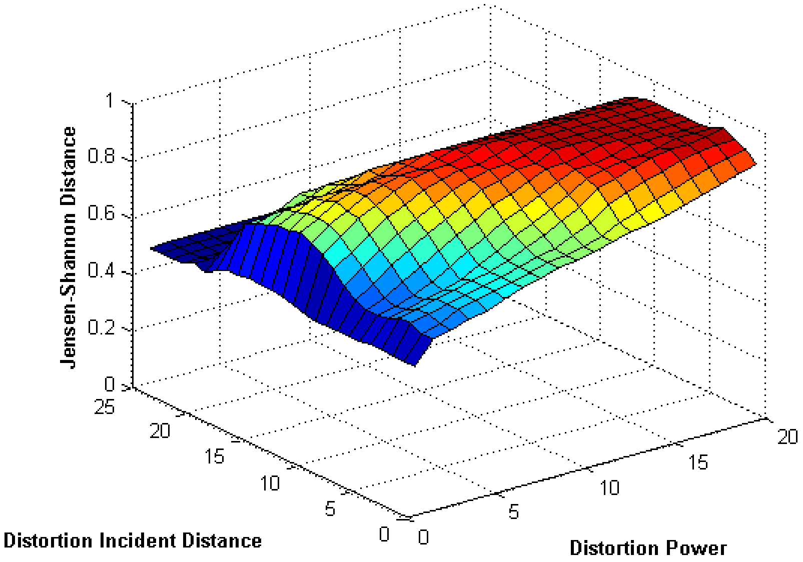


Figure 5.71: Path 17 Bias DDLHM JSds

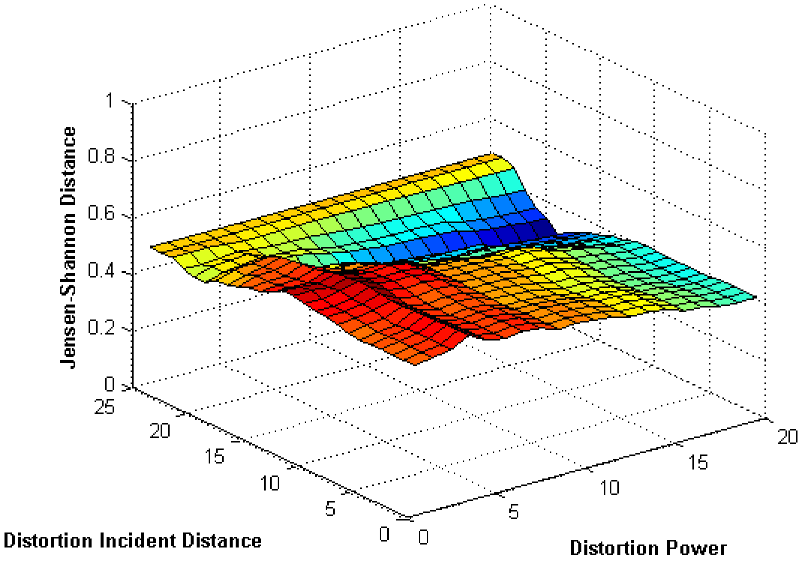


Figure 5.72: Path 17 Multipath DDLHM JSds

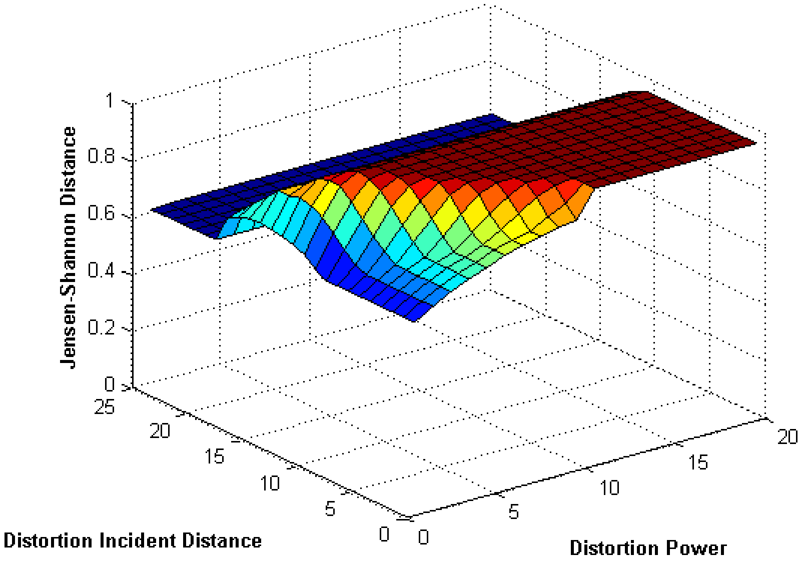


Figure 5.73: Path 18 Attenuation DDLHM JSds

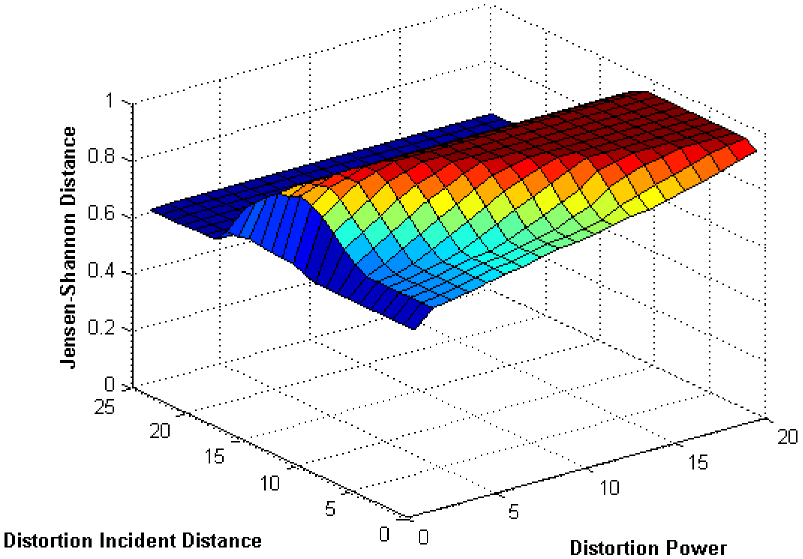


Figure 5.74: Path 18 Bias DDLHM JSds

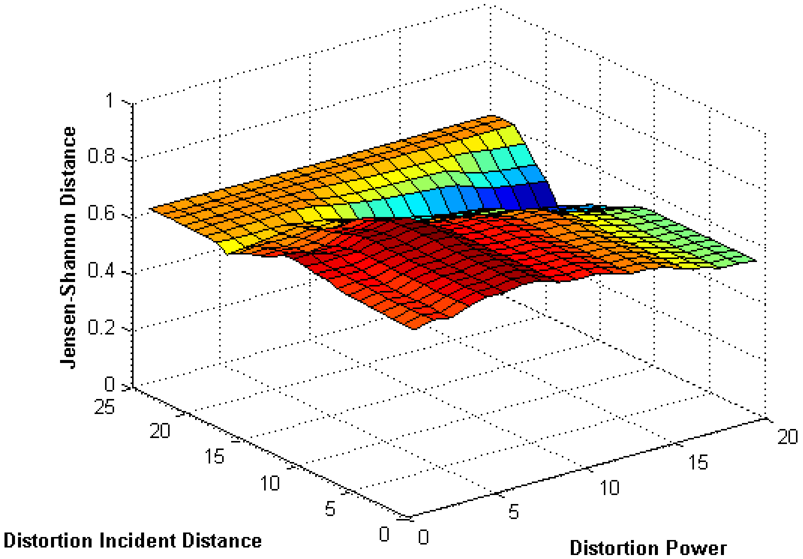


Figure 5.75: Path 18 Multipath DDLHM JSds

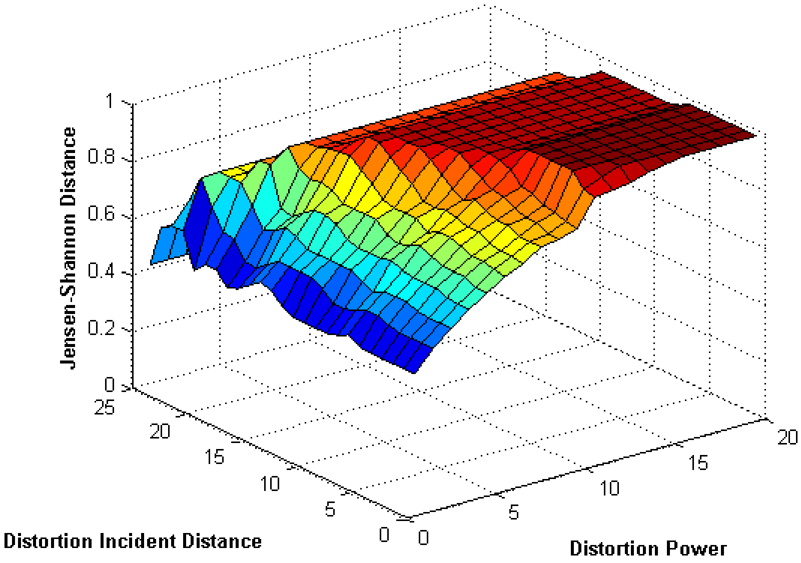


Figure 5.76: Path 19 Attenuation DDLHM JSds

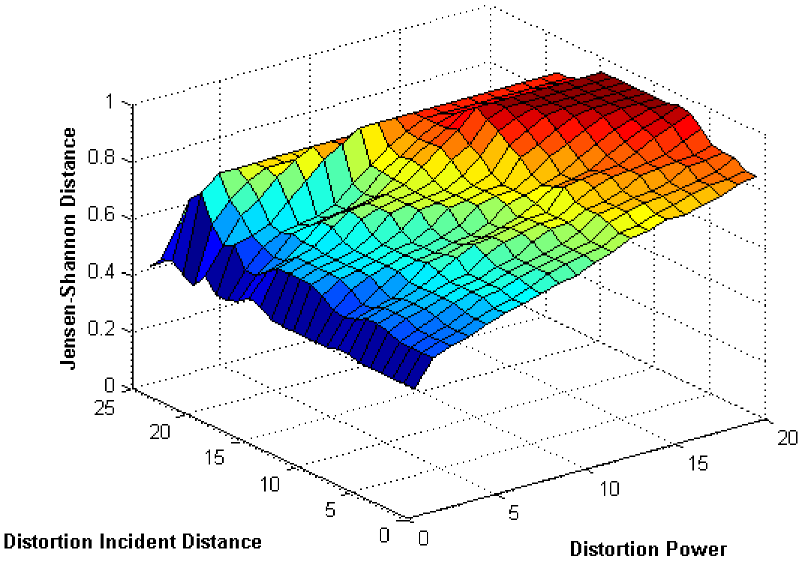


Figure 5.77: Path 19 Bias DDLHM JSds

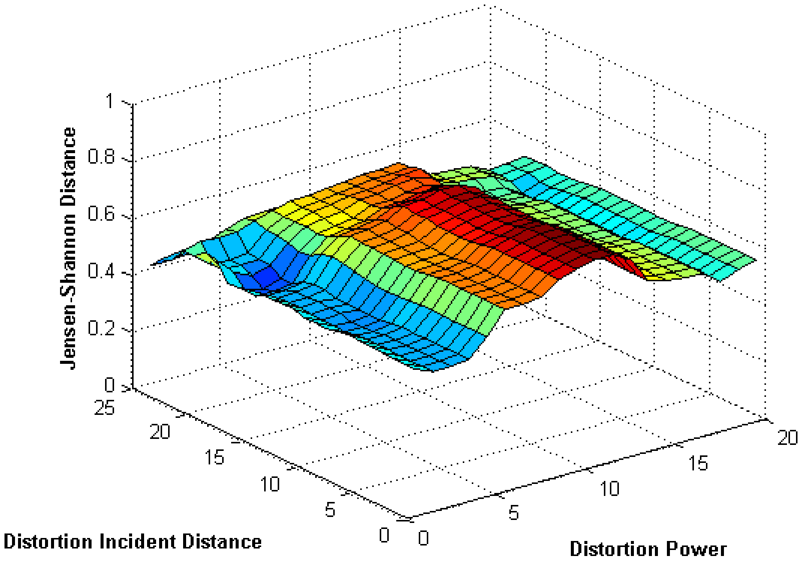


Figure 5.78: Path 19 Multipath DDLHM JSds

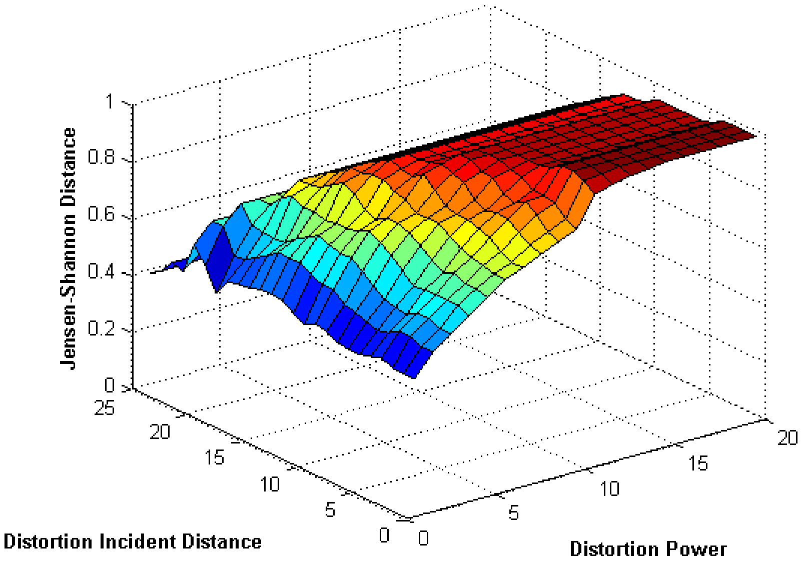


Figure 5.79: Path 20 Attenuation DDLHM JSds

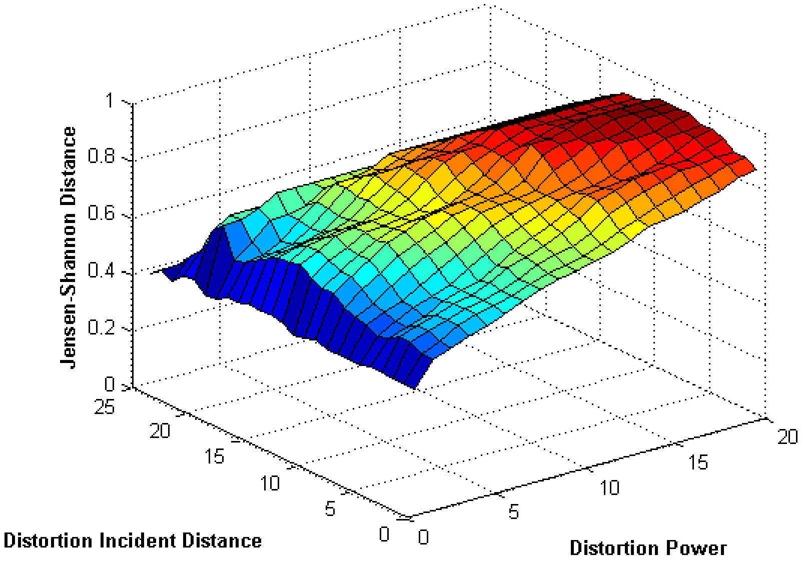


Figure 5.80: Path 20 Bias DDLHM JSds

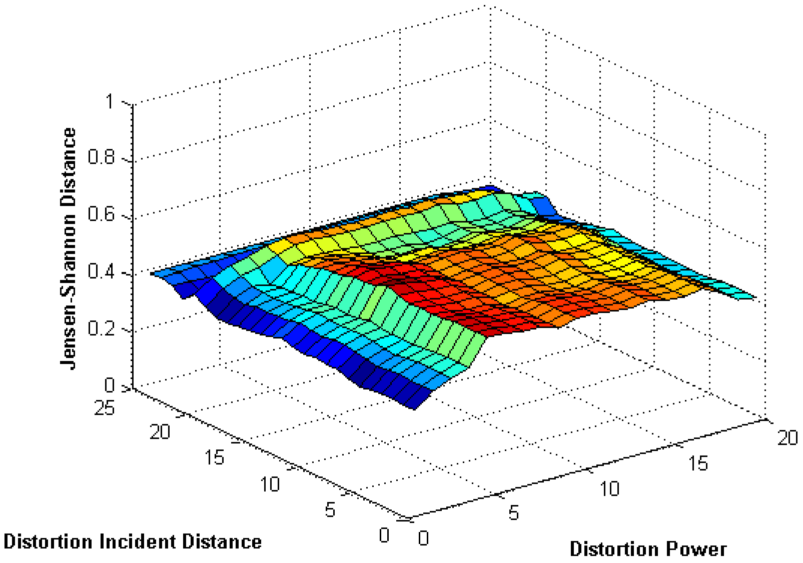


Figure 5.81: Path 20 Multipath DDLHM JSds

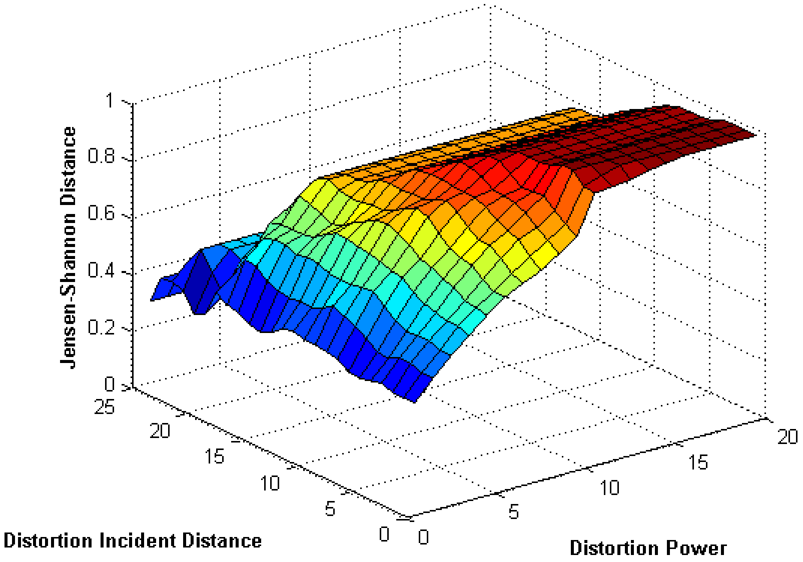


Figure 5.82: Path 21 Attenuation DDLHM JSds

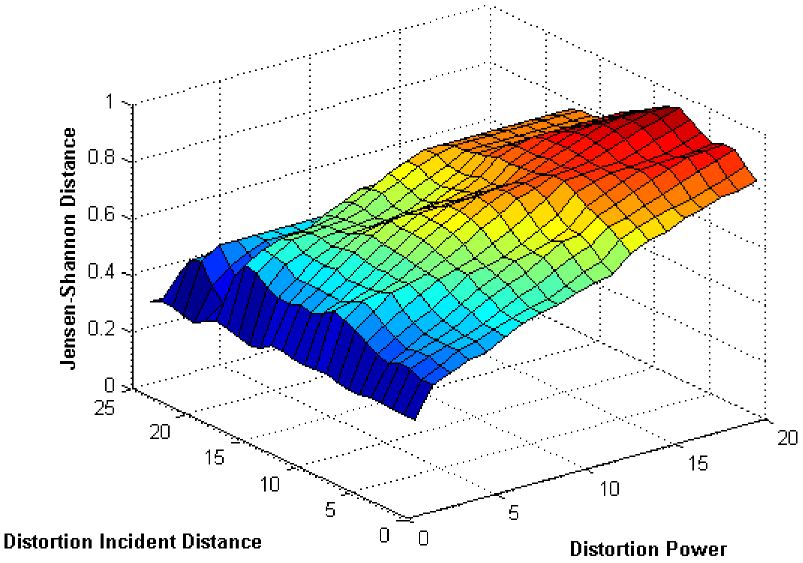


Figure 5.83: Path 21 Bias DDLHM JSds

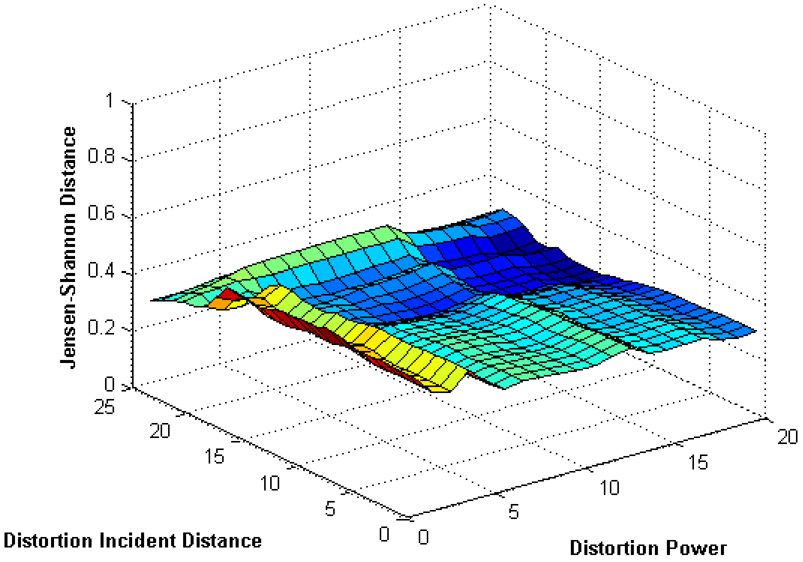


Figure 5.84: Path 21 Multipath DDLHM JSds

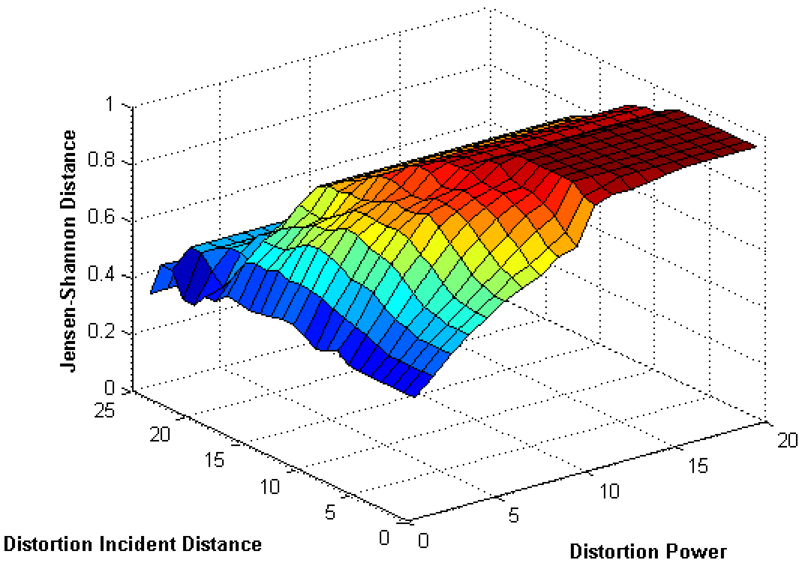


Figure 5.85: Path 22 Attenuation DDLHM JSds

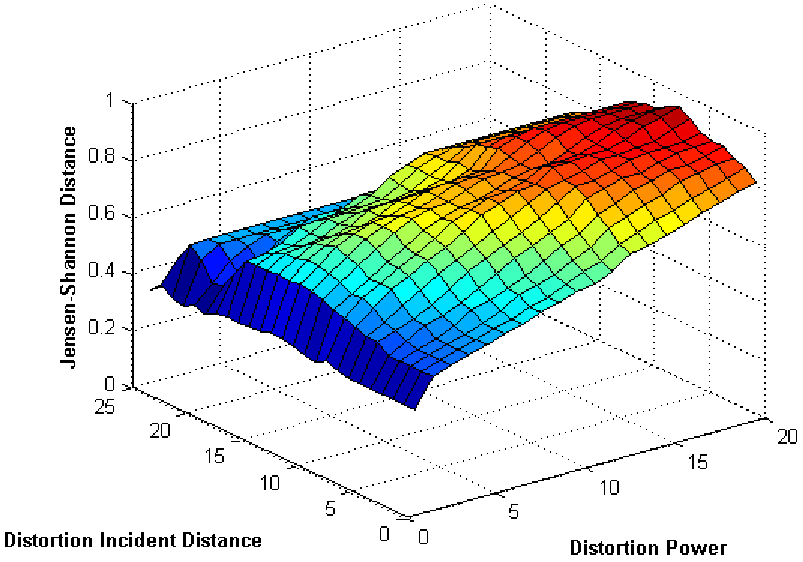


Figure 5.86: Path 22 Bias DDLHM JSds

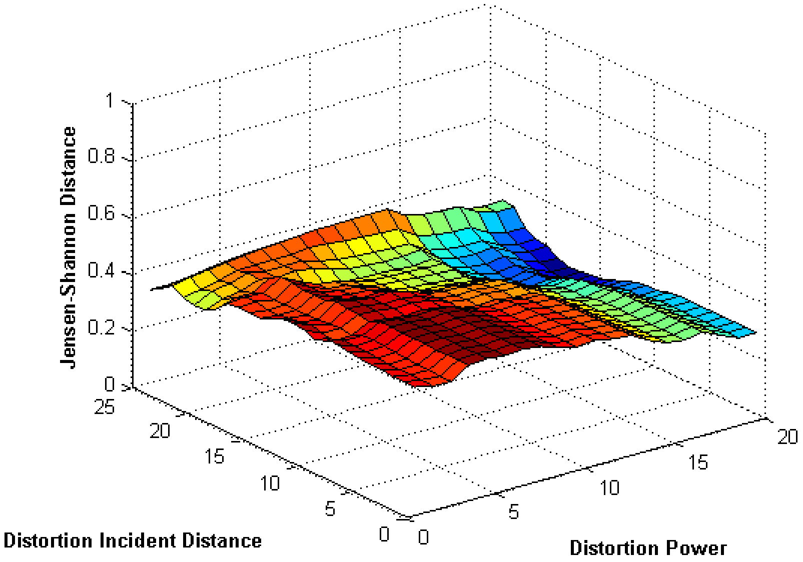


Figure 5.87: Path 22 Multipath DDLHM JSds

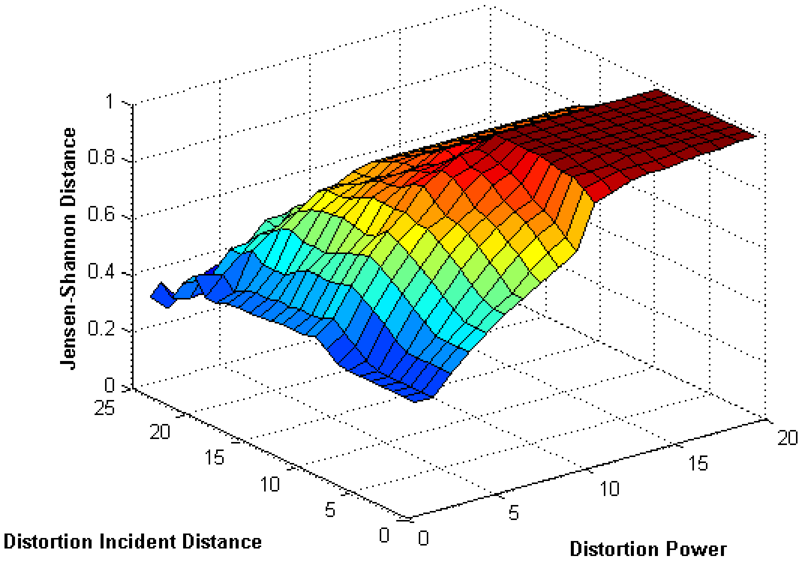


Figure 5.88: Path 23 Attenuation DDLHM JSds

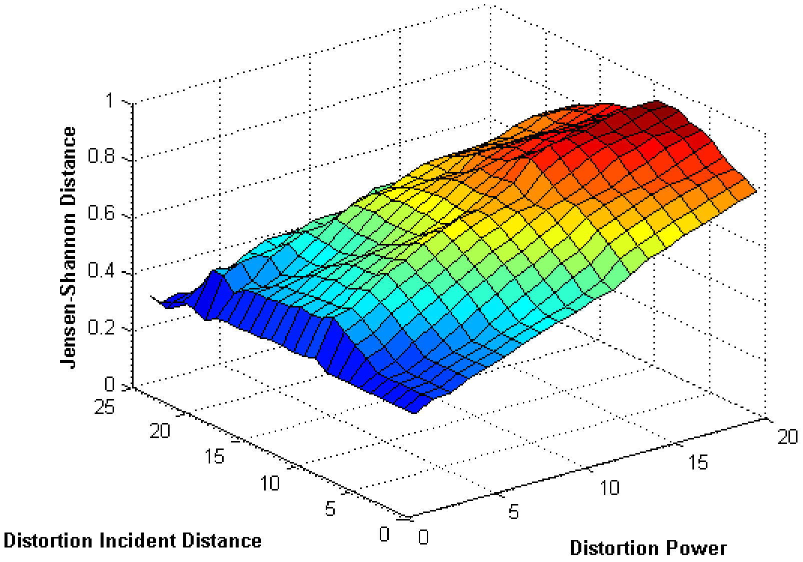


Figure 5.89: Path 23 Bias DDLHM JSds

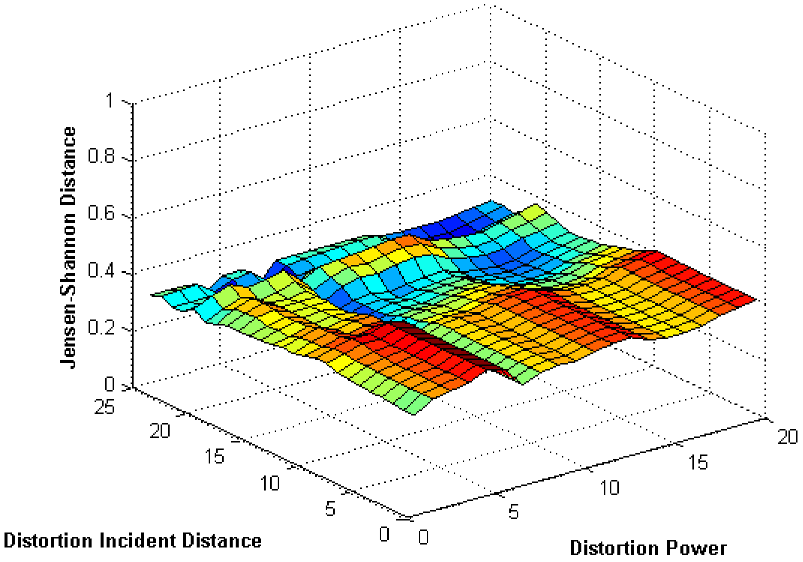


Figure 5.90: Path 23 Multipath DDLHM JSds

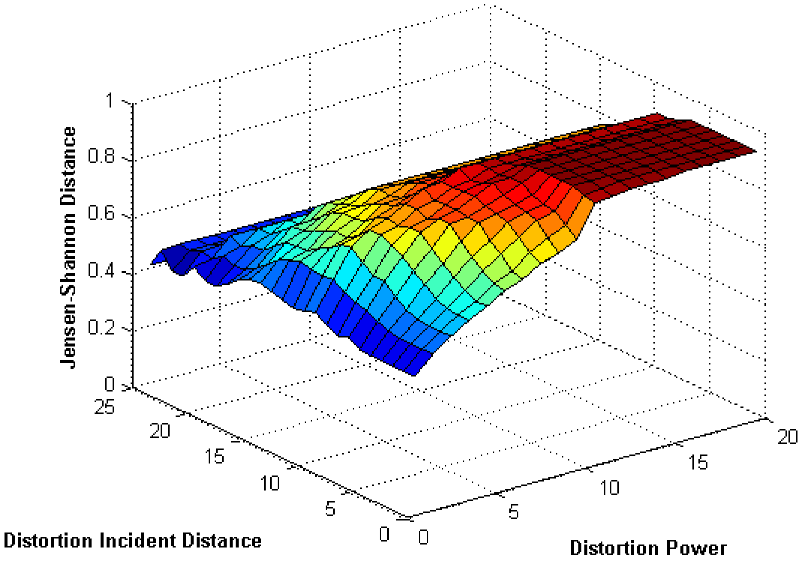


Figure 5.91: Path 24 Attenuation DDLHM JSds

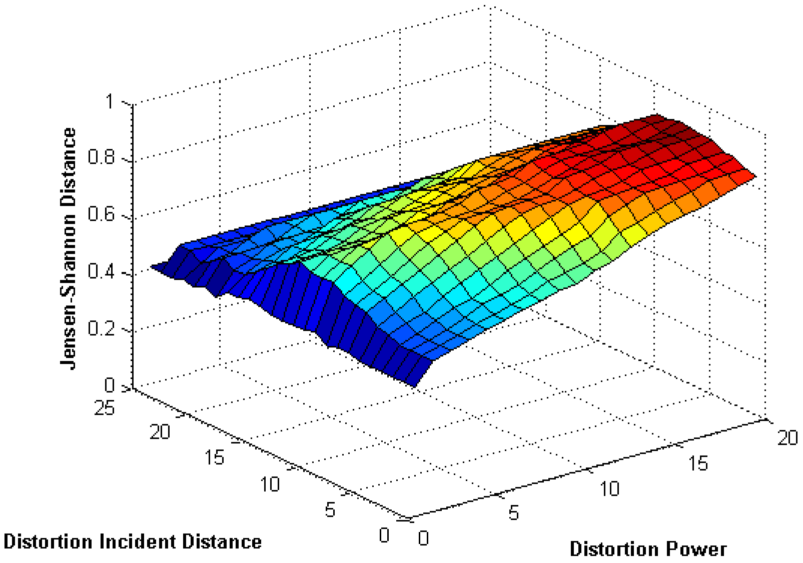


Figure 5.92: Path 24 Bias DDLHM JSds

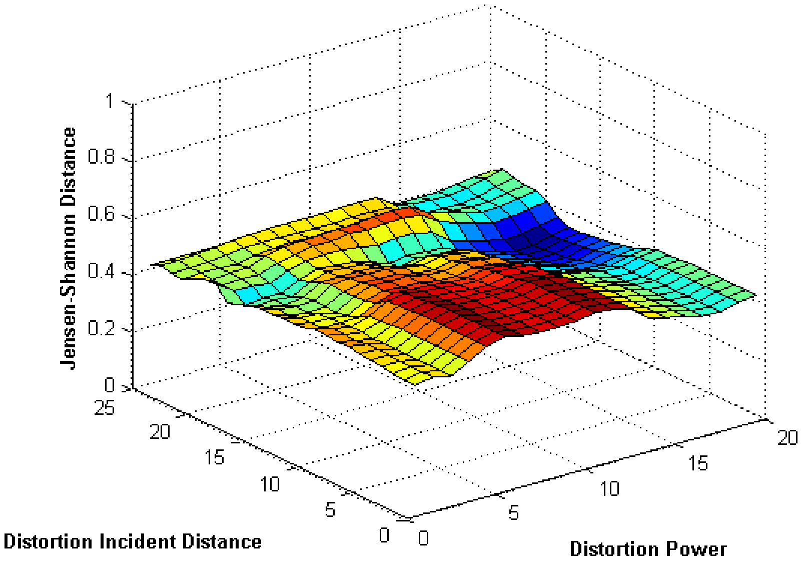


Figure 5.93: Path 24 Multipath DDLHM JSds

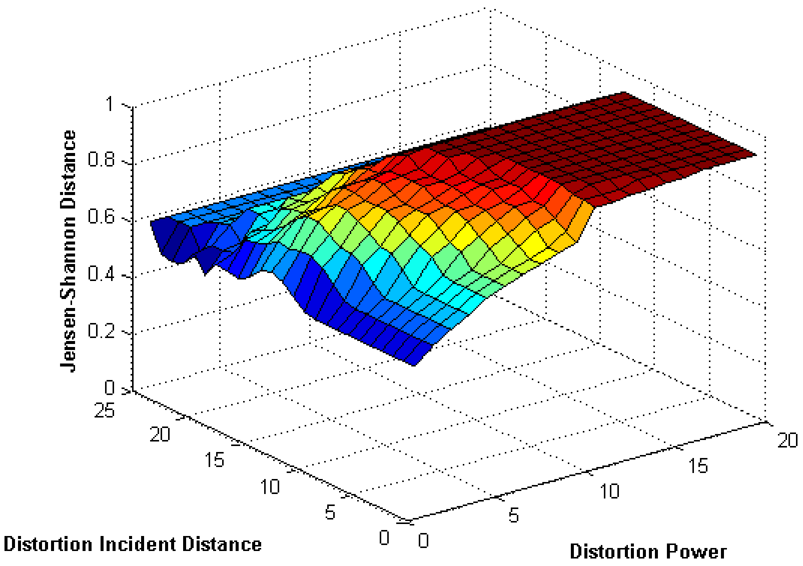


Figure 5.94: Path 25 Attenuation DDLHM JSds

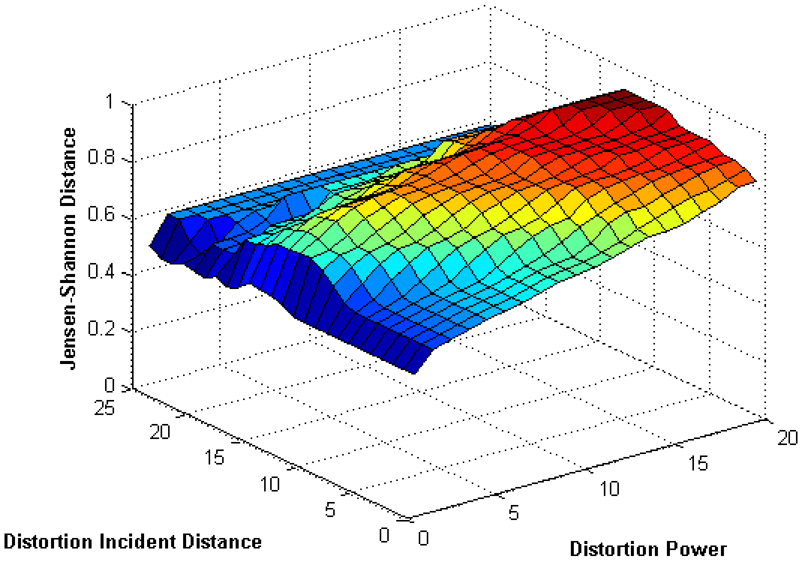


Figure 5.95: Path 25 Bias DDLHM JSds

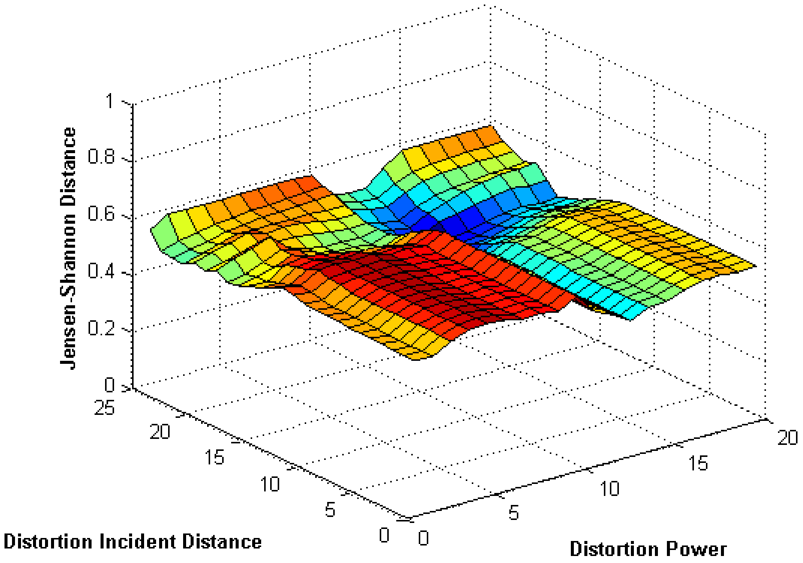


Figure 5.96: Path 25 Multipath DDLHM JSds

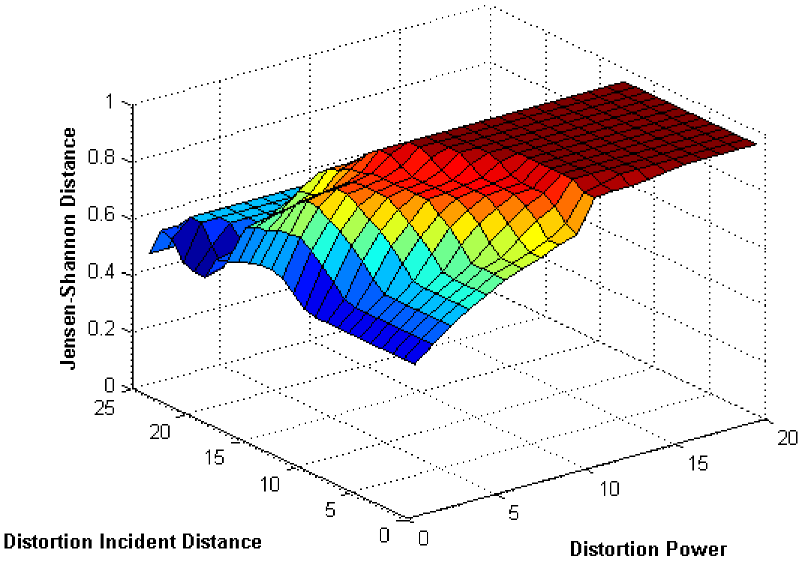


Figure 5.97: Path 26 Attenuation DDLHM JSds

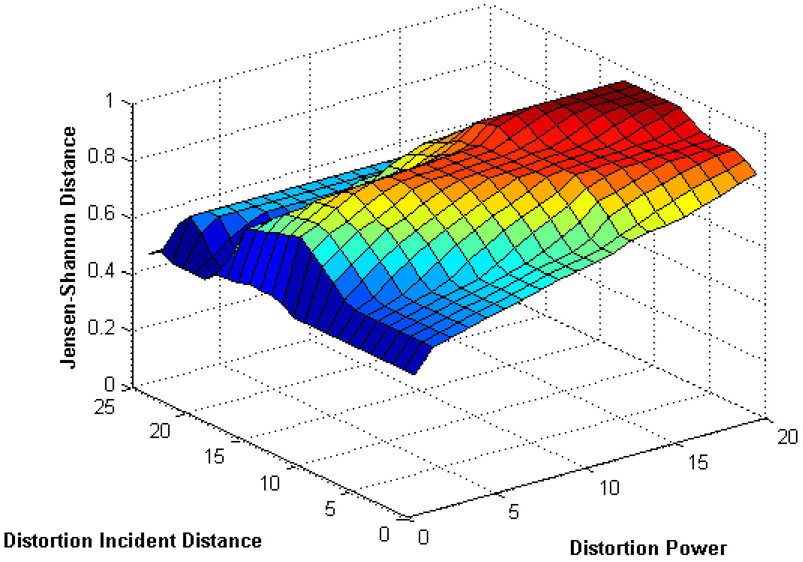


Figure 5.98: Path 26 Bias DDLHM JSds

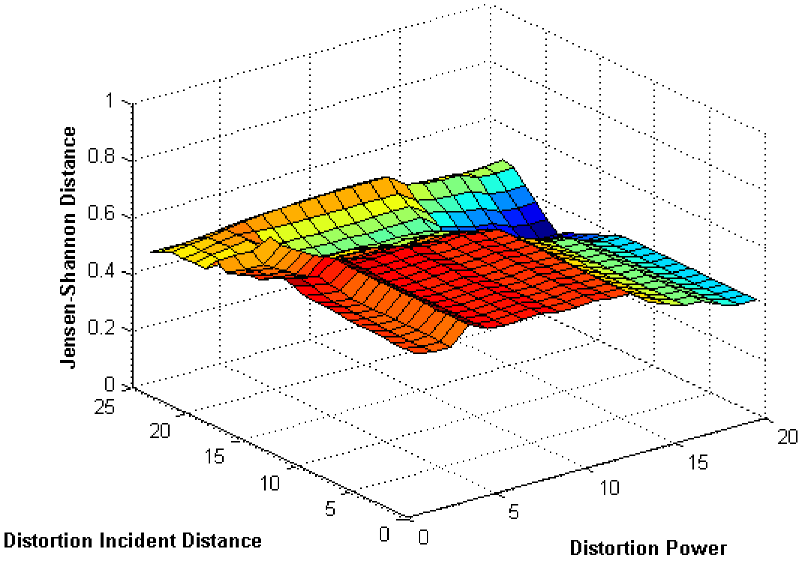


Figure 5.99: Path 26 Multipath DDLHM JSds

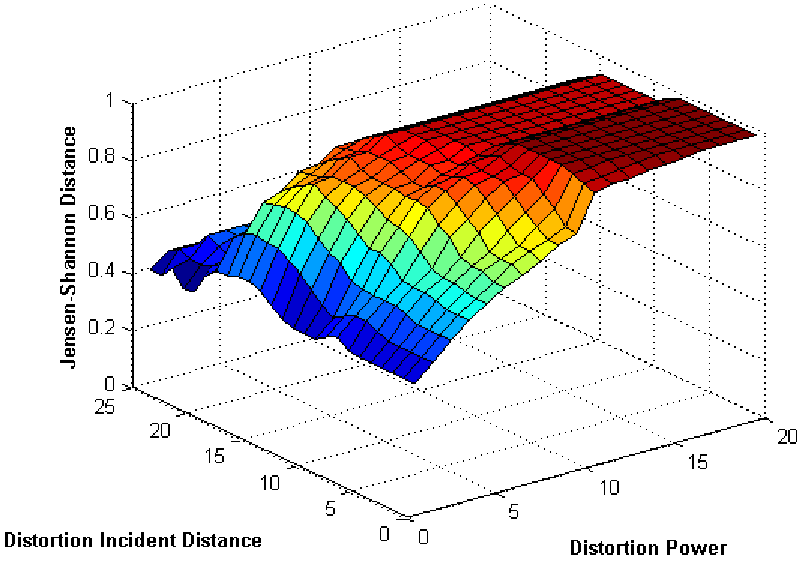


Figure 5.100: Path 27 Attenuation DDLHM JSds

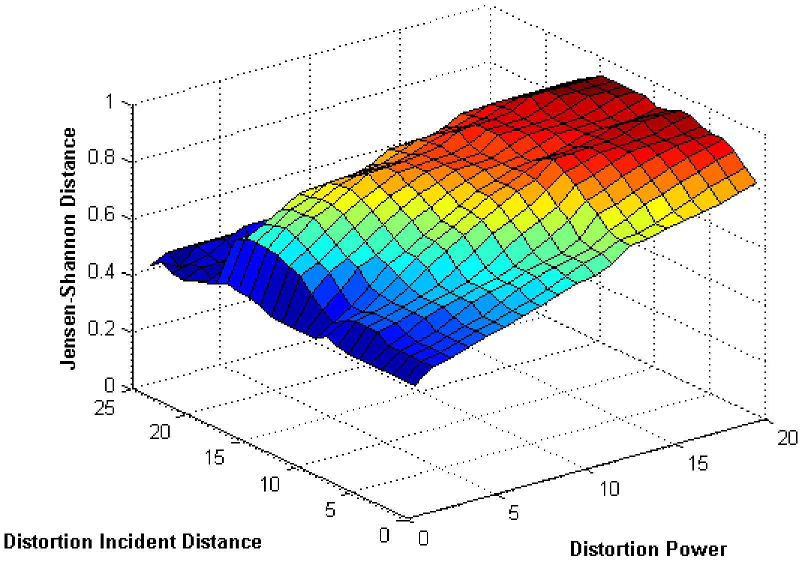


Figure 5.101: Path 27 Bias DDLHM JSds

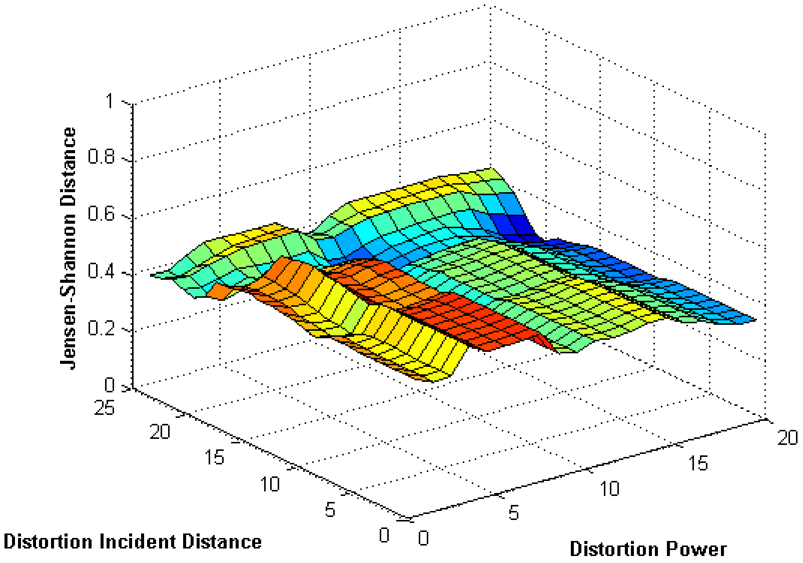


Figure 5.102: Path 27 Multipath DDLHM JSds

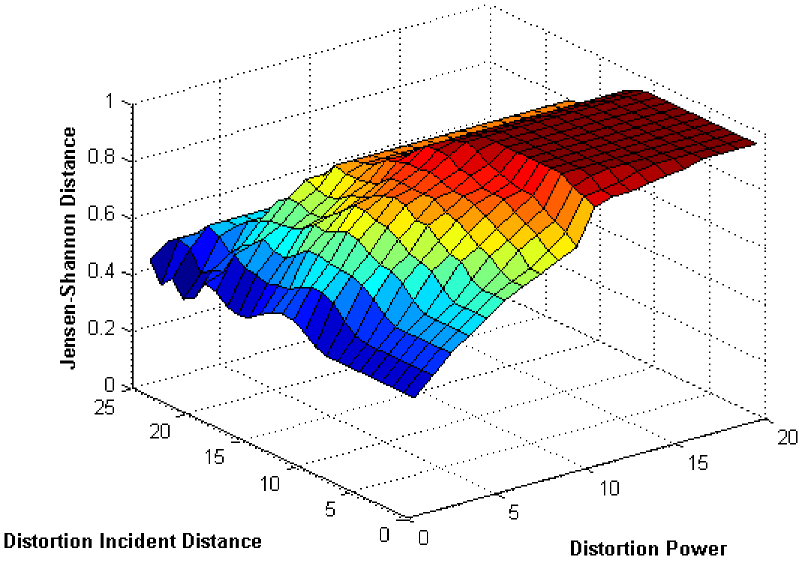


Figure 5.103: Path 28 Attenuation DDLHM JSds

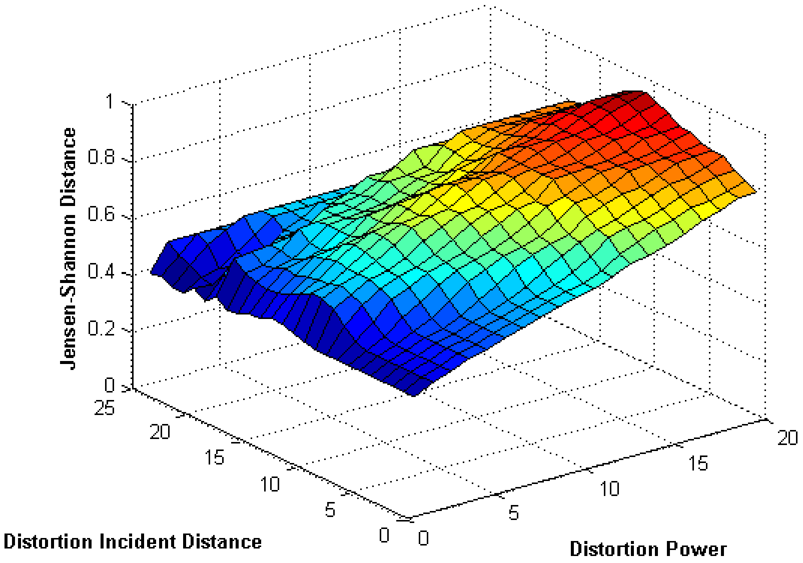


Figure 5.104: Path 28 Bias DDLHM JSds

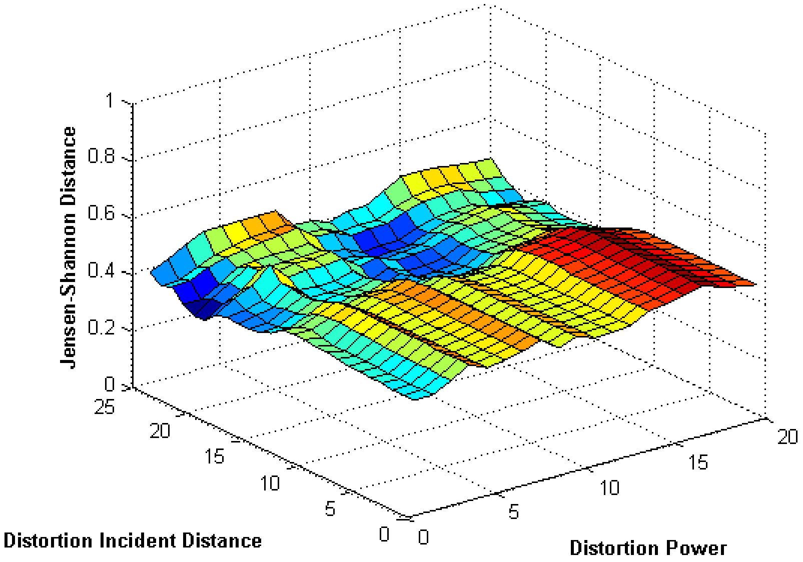


Figure 5.105: Path 28 Multipath DDLHM JSds

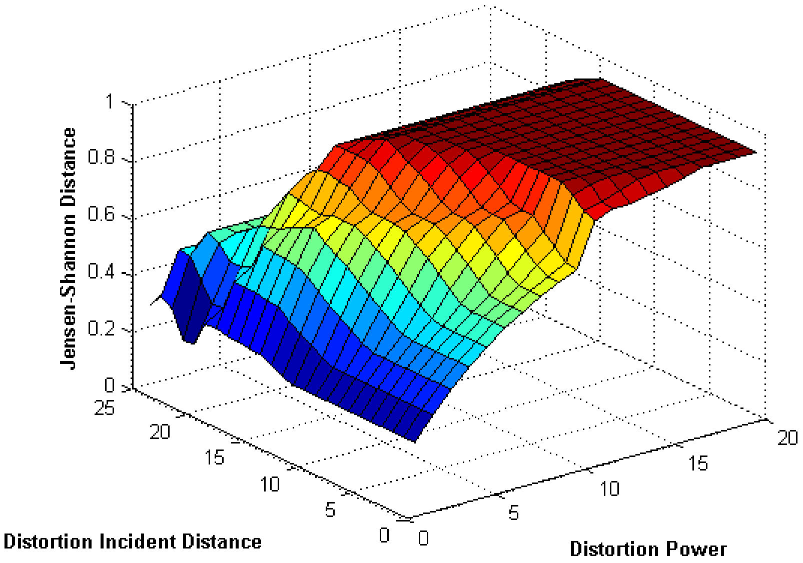


Figure 5.106: Path 29 Attenuation DDLHM JSds

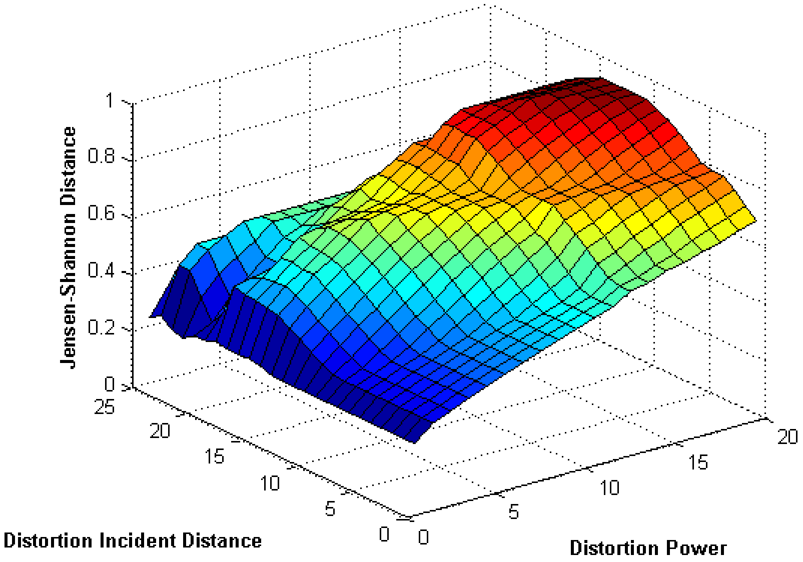


Figure 5.107: Path 29 Bias DDLHM JSds

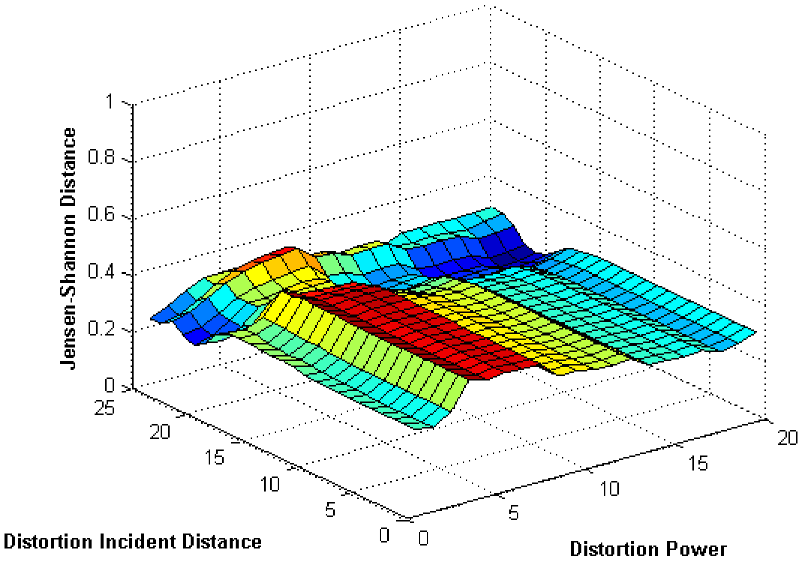


Figure 5.108: Path 29 Multipath DDLHM JSds

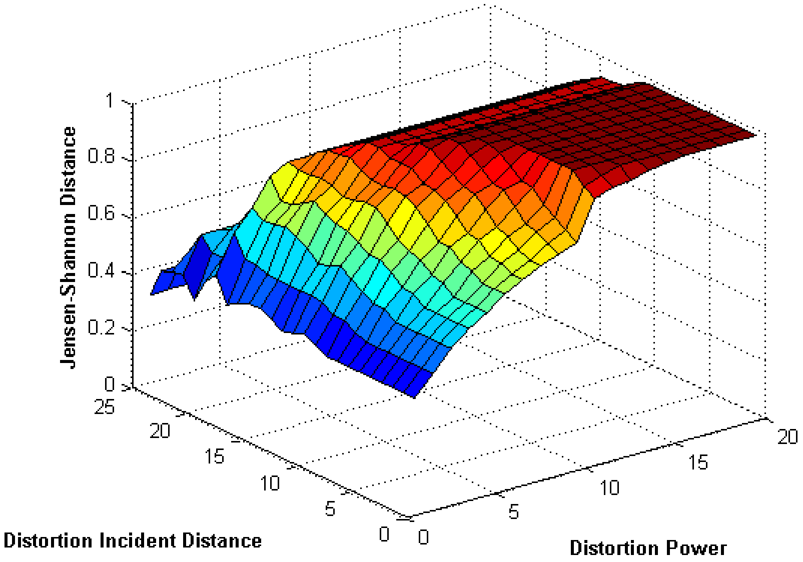


Figure 5.109: Path 30 Attenuation DDLHM JSds

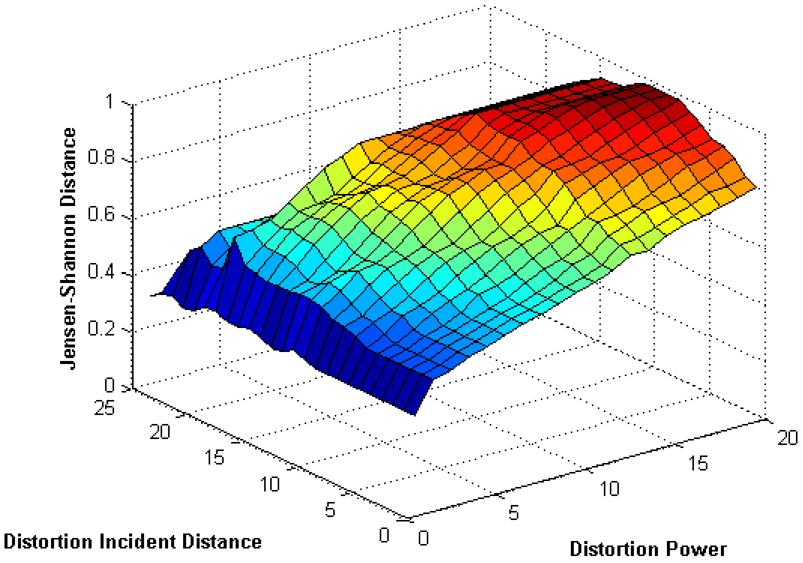


Figure 5.110: Path 30 Bias DDLHM JSds

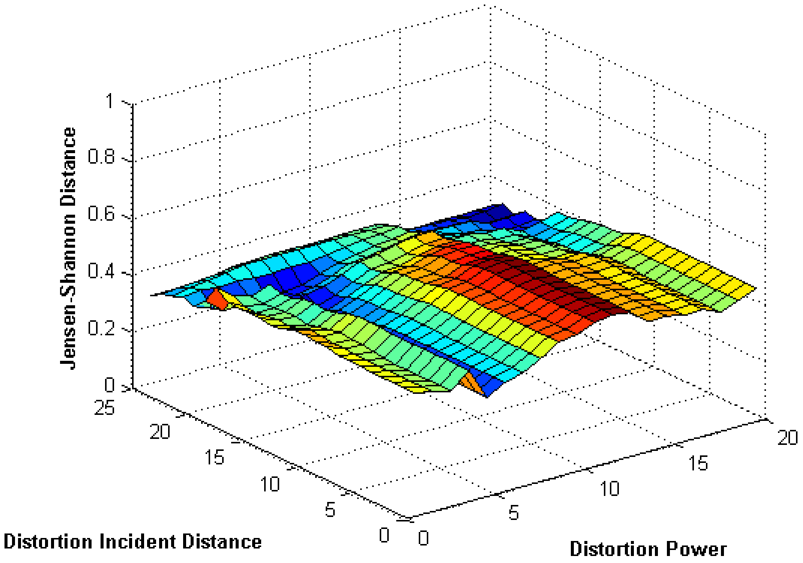


Figure 5.111: Path 30 Multipath DDLHM JSds

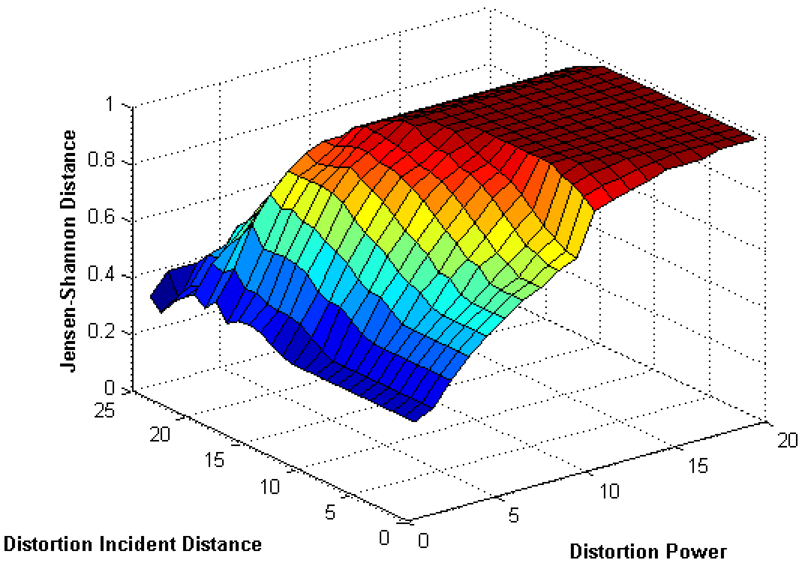


Figure 5.112: Path 31 Attenuation DDLHM JSds

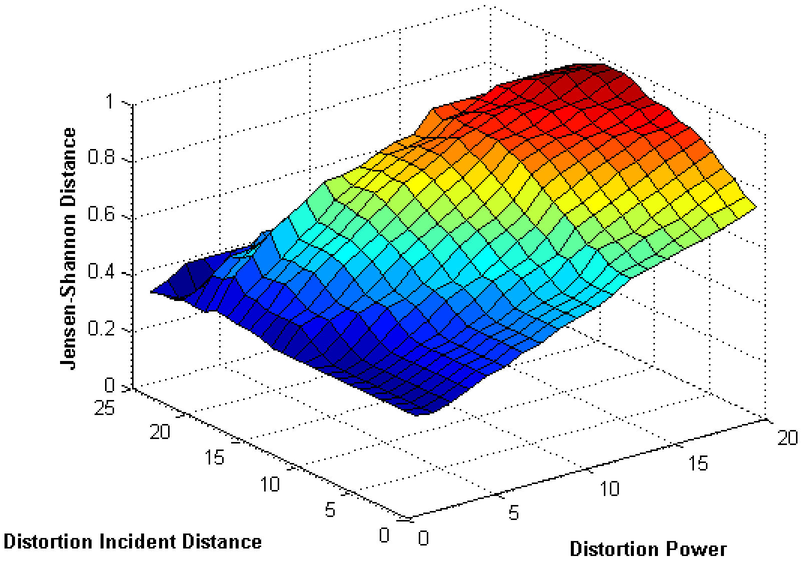


Figure 5.113: Path 31 Bias DDLHM JSds

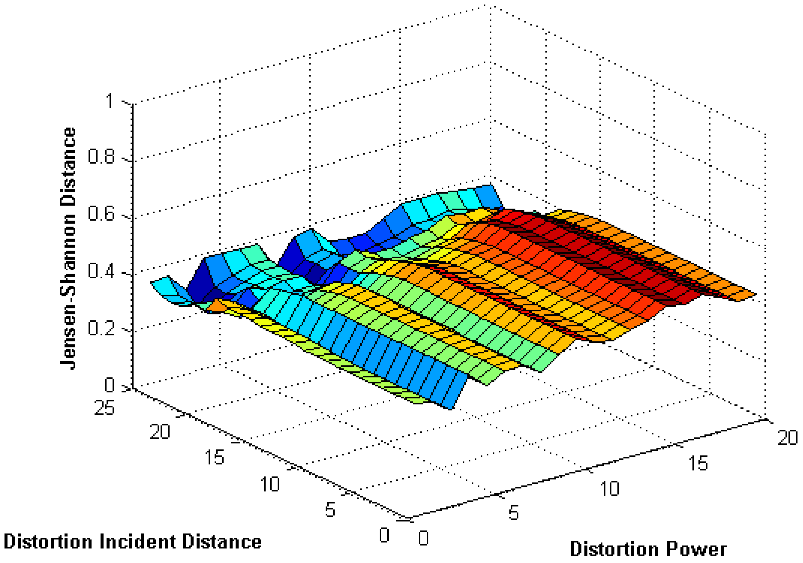


Figure 5.114: Path 31 Multipath DDLHM JSds

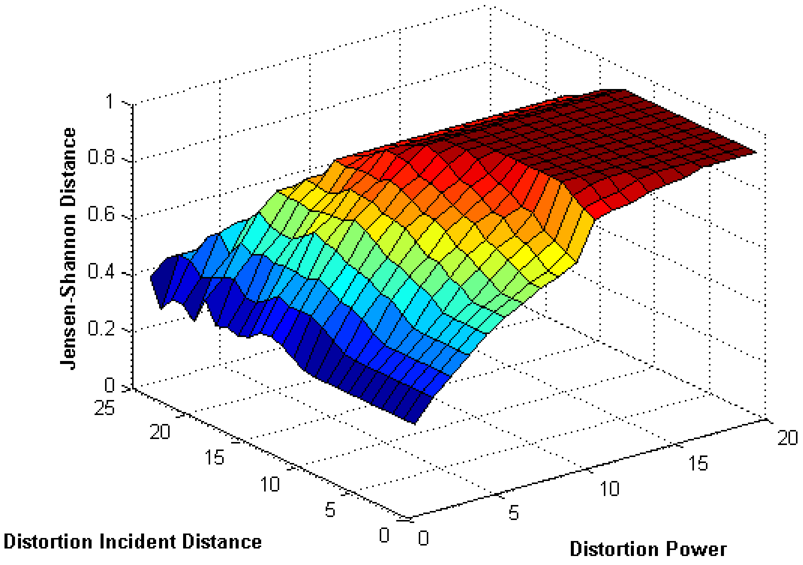


Figure 5.115: Path 32 Attenuation DDLHM JSds

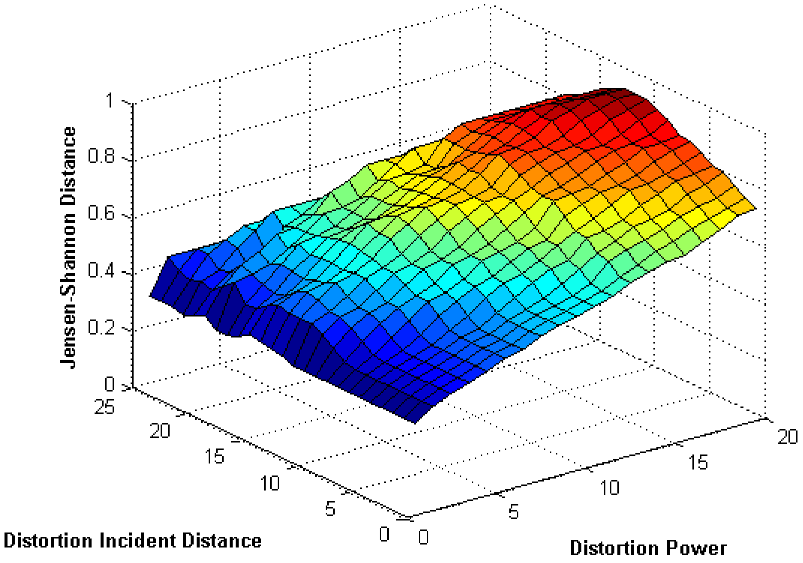


Figure 5.116: Path 32 Bias DDLHM JSds

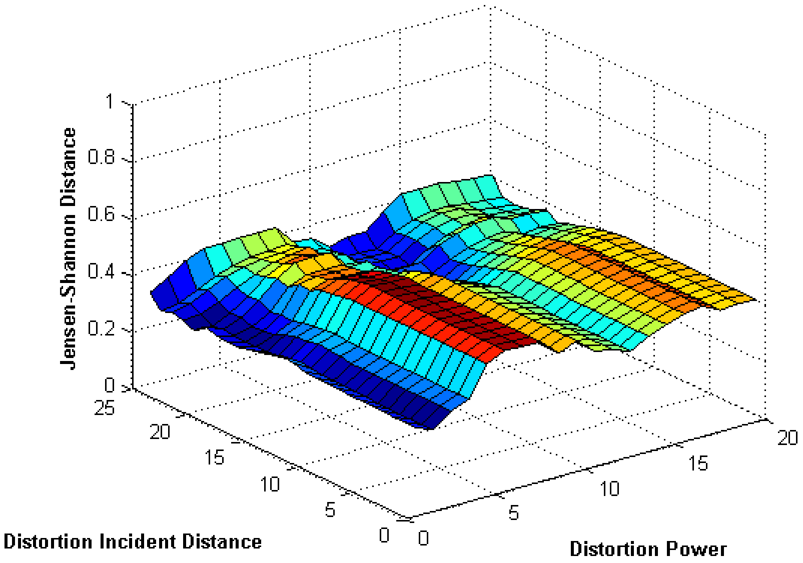


Figure 5.117: Path 32 Multipath DDLHM JSds

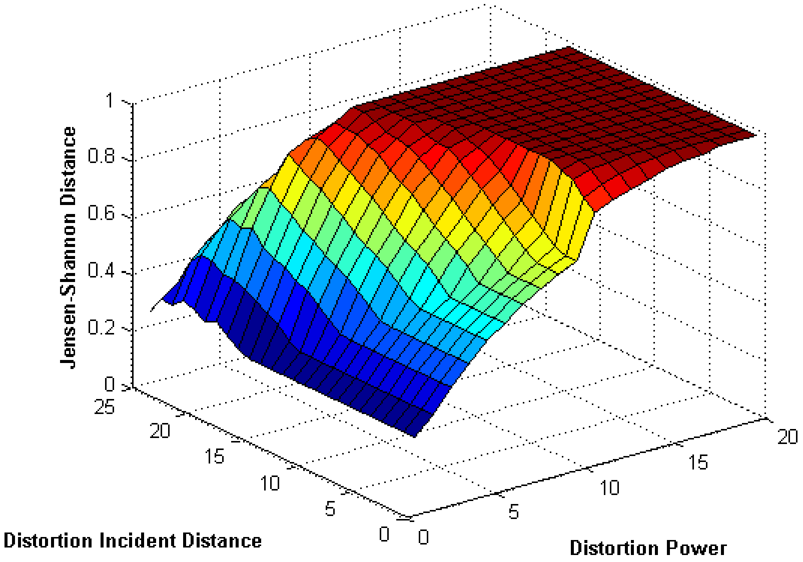


Figure 5.118: Path 33 Attenuation DDLHM JSds

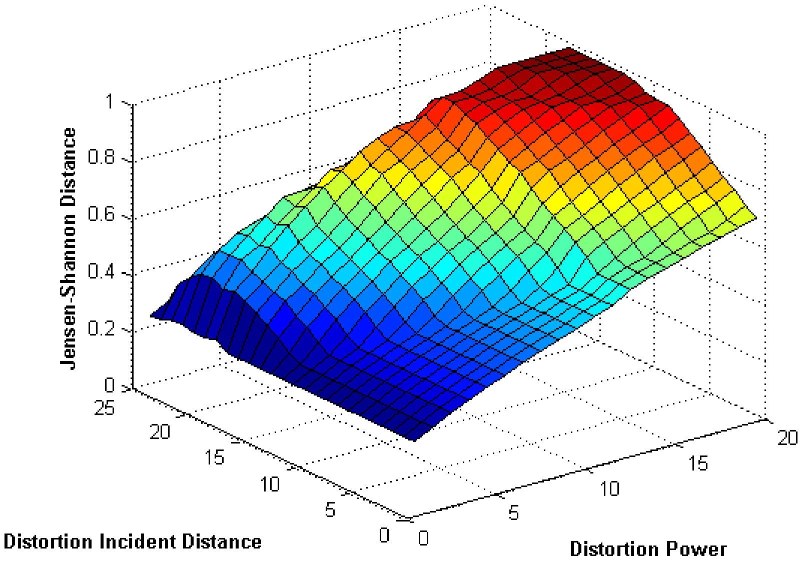


Figure 5.119: Path 33 Bias DDLHM JSds

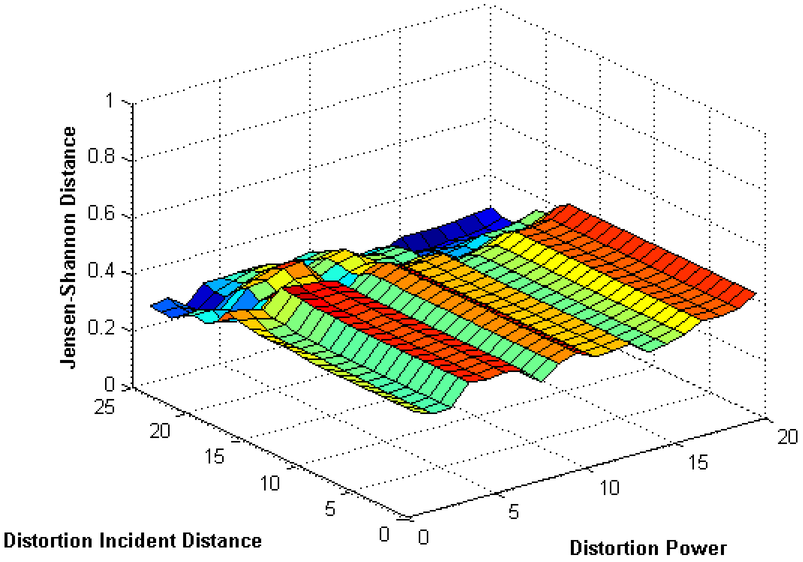


Figure 5.120: Path 33 Multipath DDLHM JSds

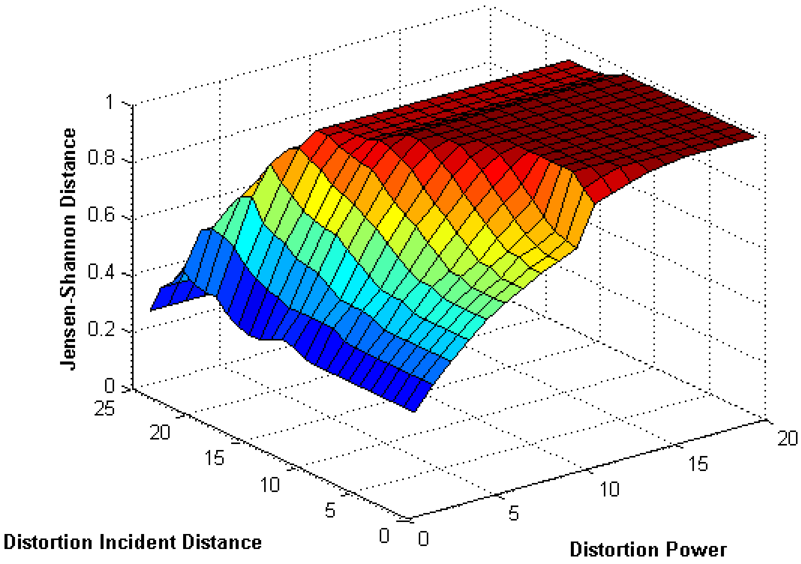


Figure 5.121: Path 34 Attenuation DDLHM JSds

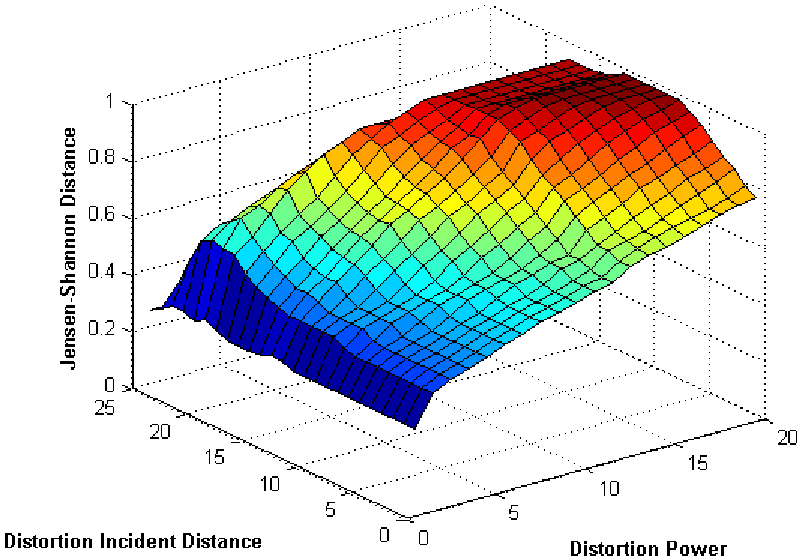


Figure 5.122: Path 34 Bias DDLHM JSds

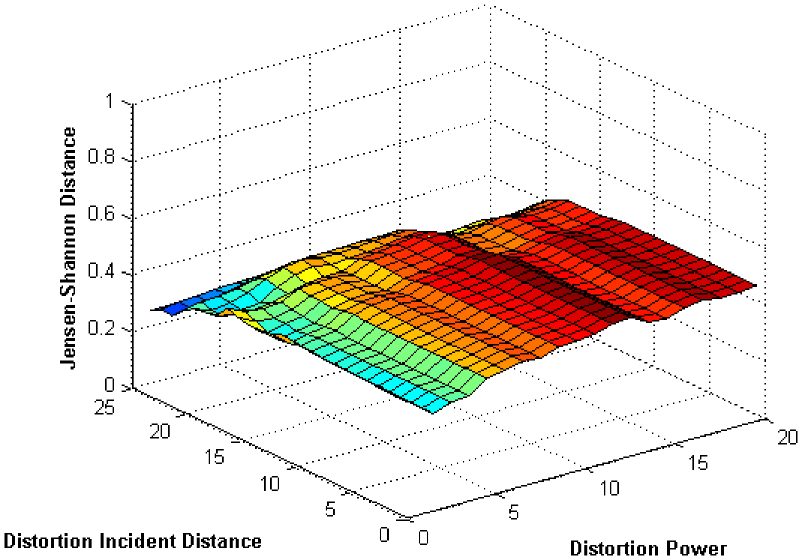


Figure 5.123: Path 34 Multipath DDLHM JSds

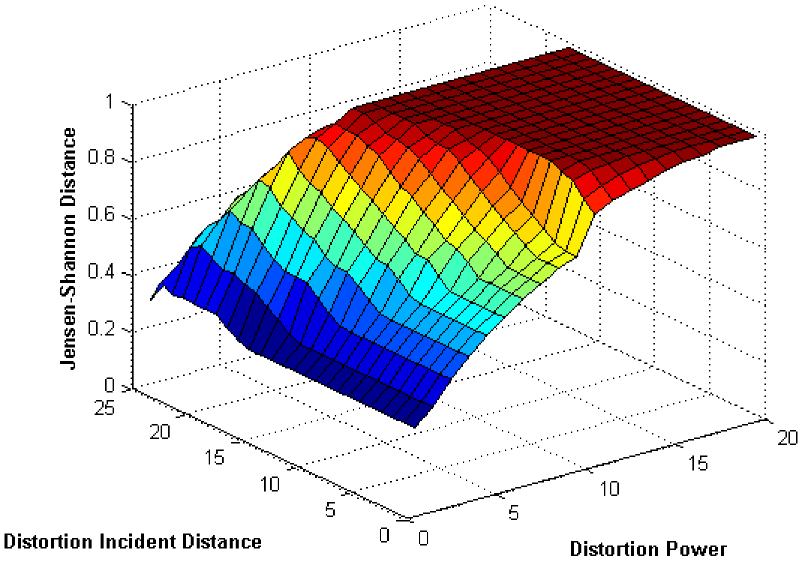


Figure 5.124: Path 35 Attenuation DDLHM JSds

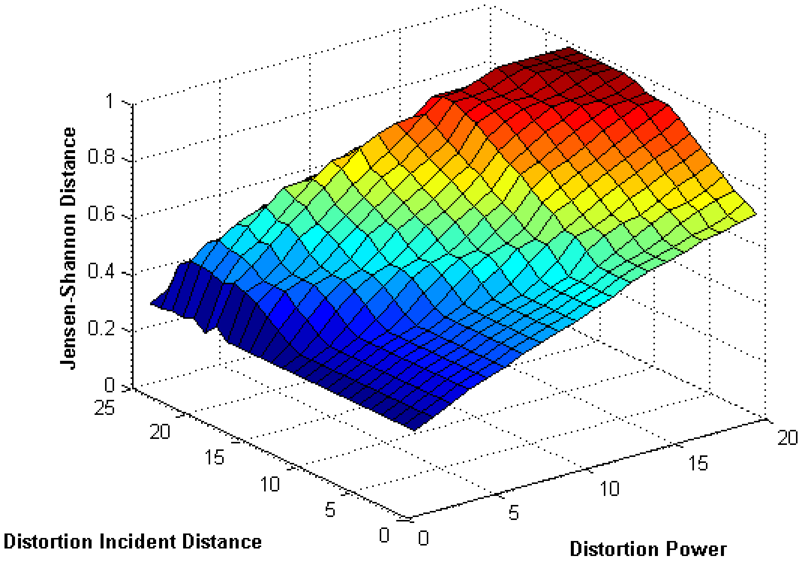


Figure 5.125: Path 35 Bias DDLHM JSds

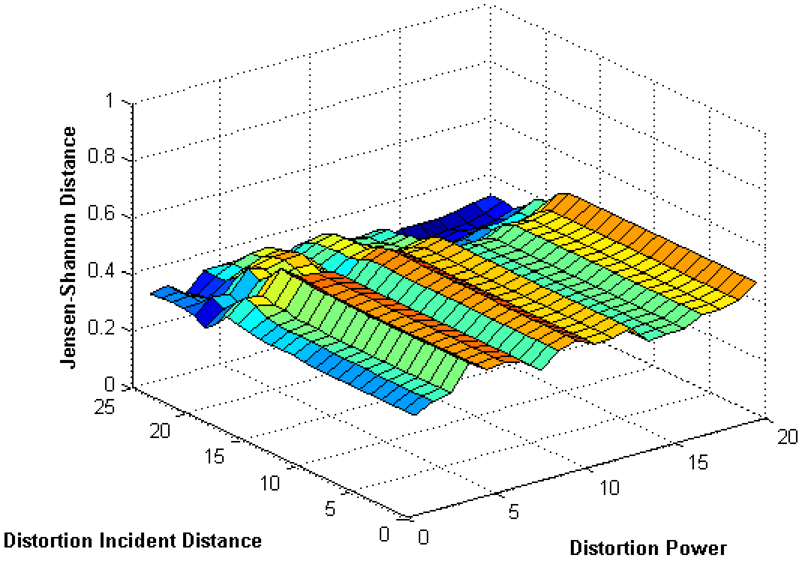


Figure 5.126: Path 35 Multipath DDLHM JSds

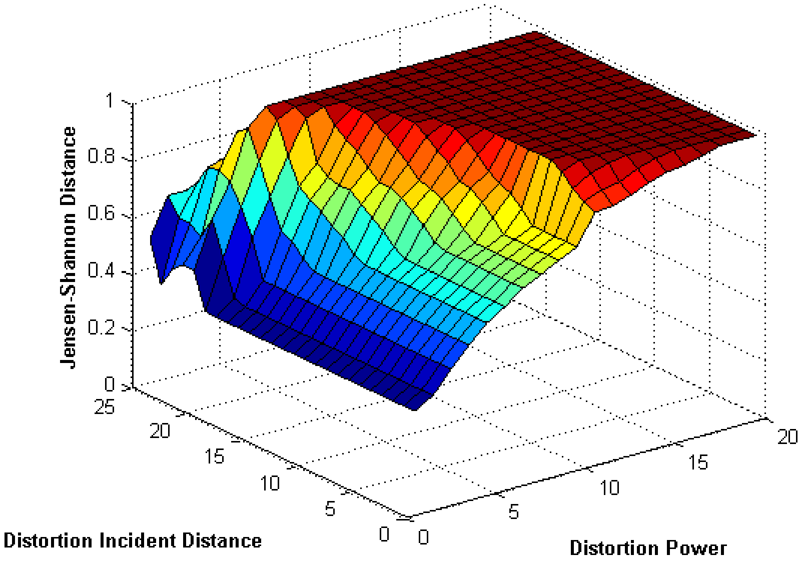


Figure 5.127: Path 36 Attenuation DDLHM JSds

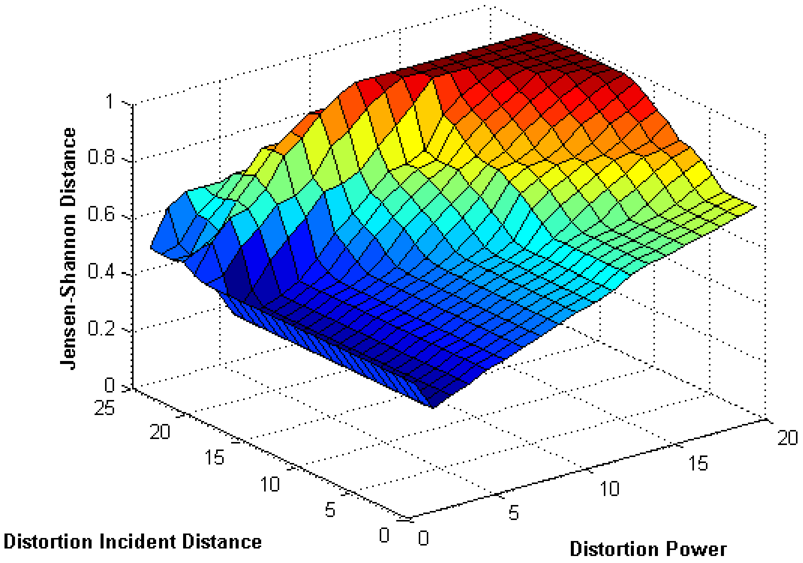


Figure 5.128: Path 36 Bias DDLHM JSds

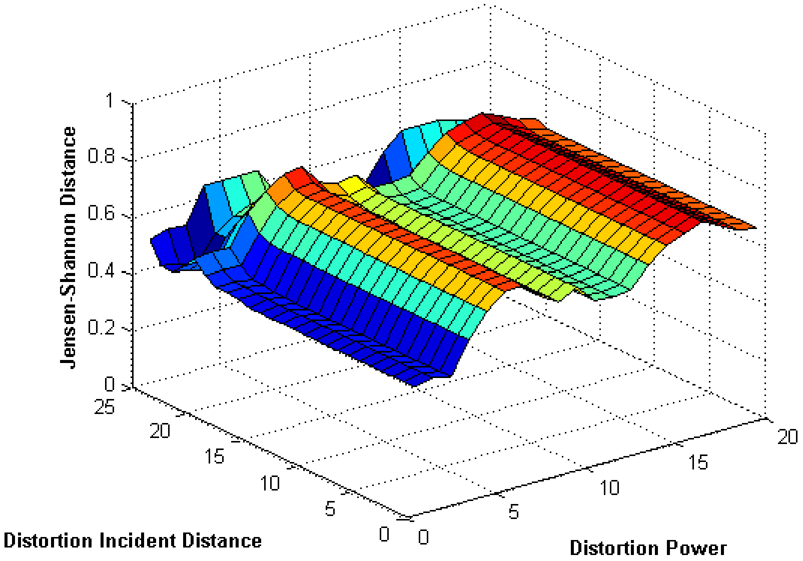


Figure 5.129: Path 36 Multipath DDLHM JSds

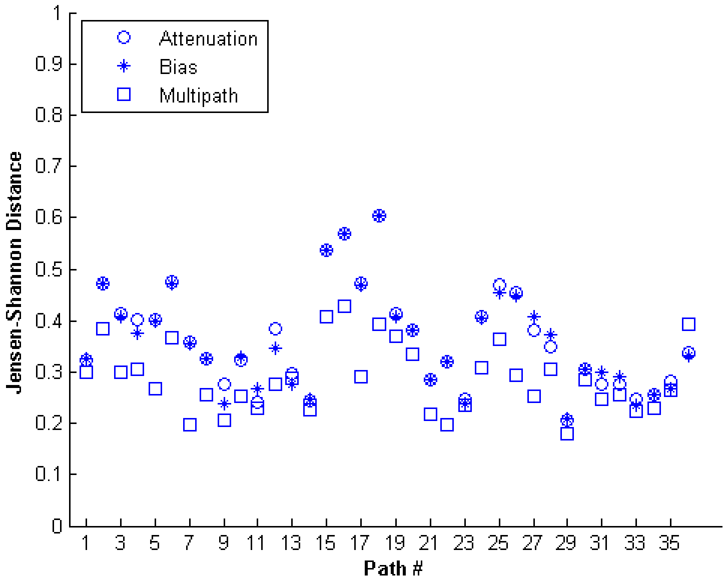


Figure 5.130: Grid Best Matches per Distortion

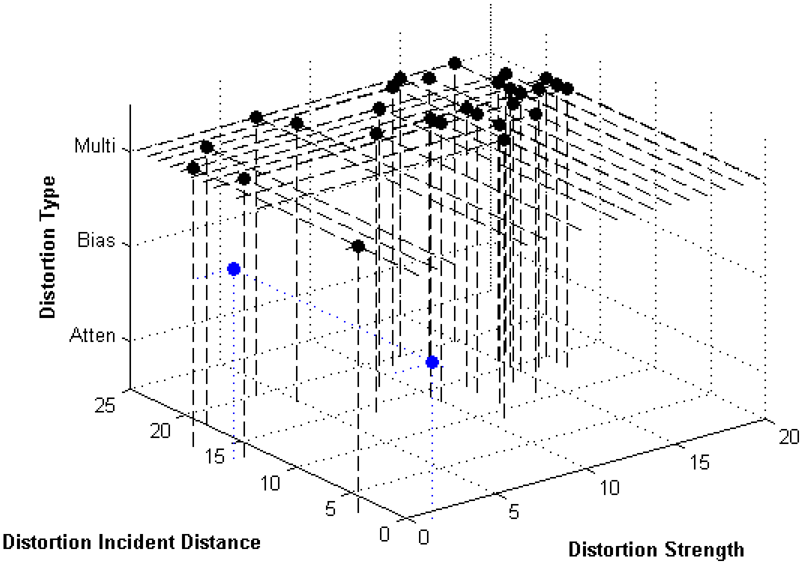


Figure 5.131: Grid Best Distortion Matches Overall

5.1.4 Distortion Characteristics of the WINLAB Environment

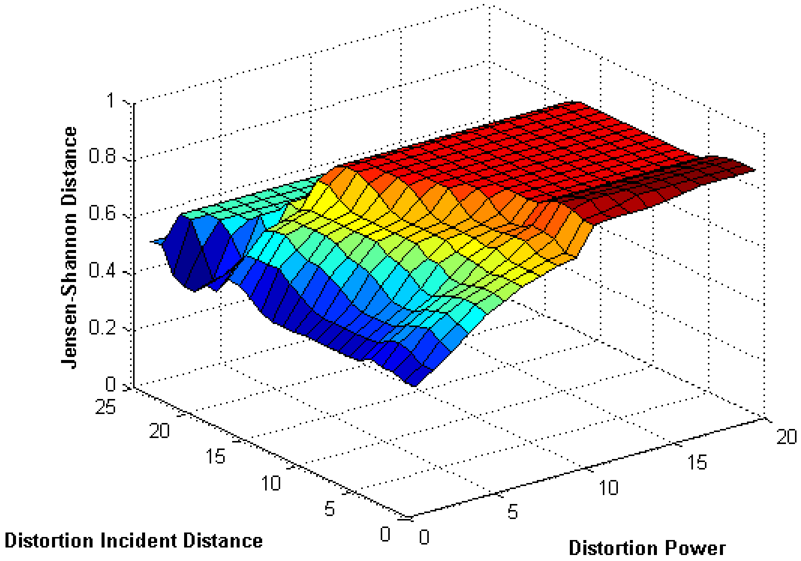


Figure 5.132: Path 1 Attenuation DDLHM JSds

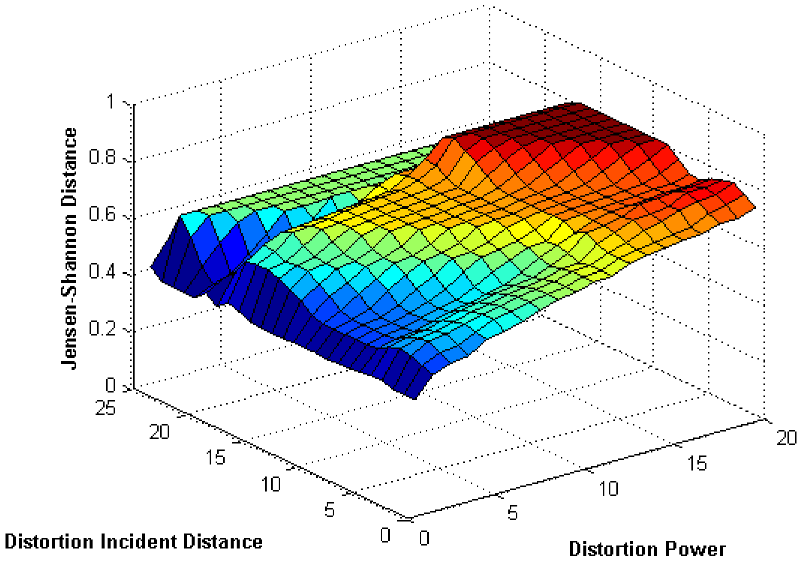


Figure 5.133: Path 1 Bias DDLHM JSds

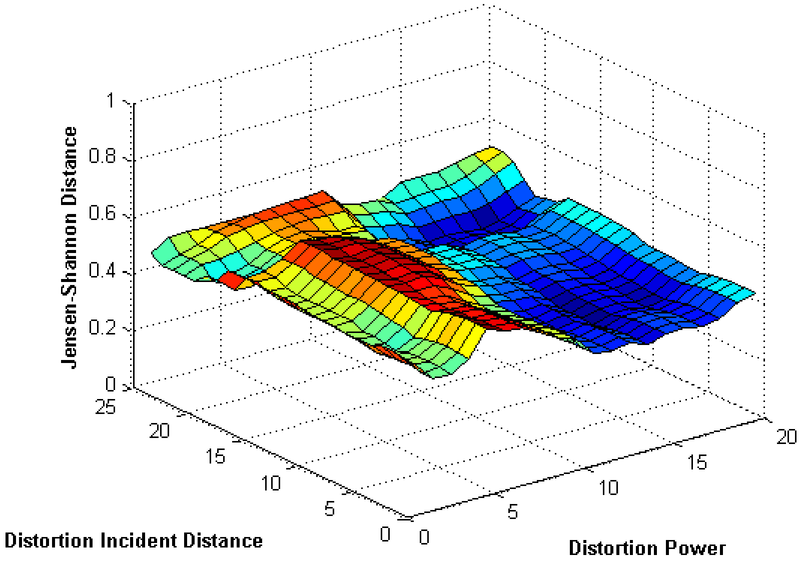


Figure 5.134: Path 1 Multipath DDLHM JSds

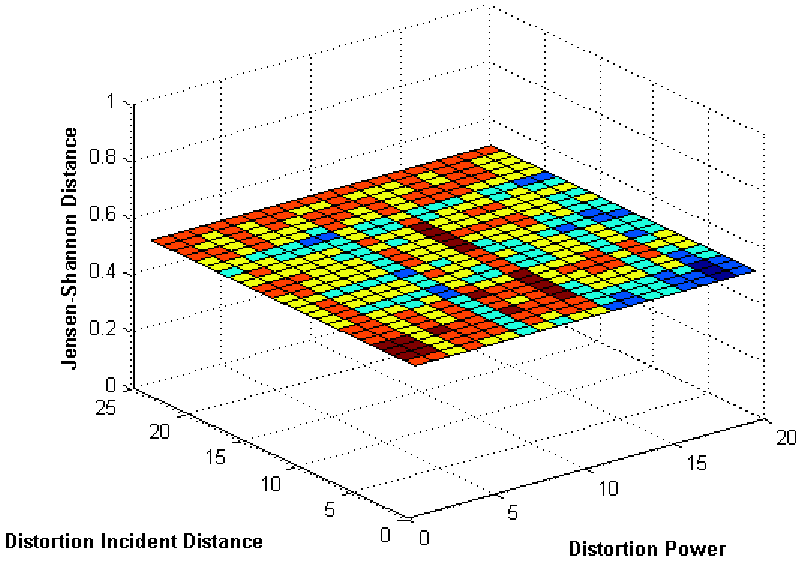


Figure 5.135: Path 2 Attenuation DDLHM JSds

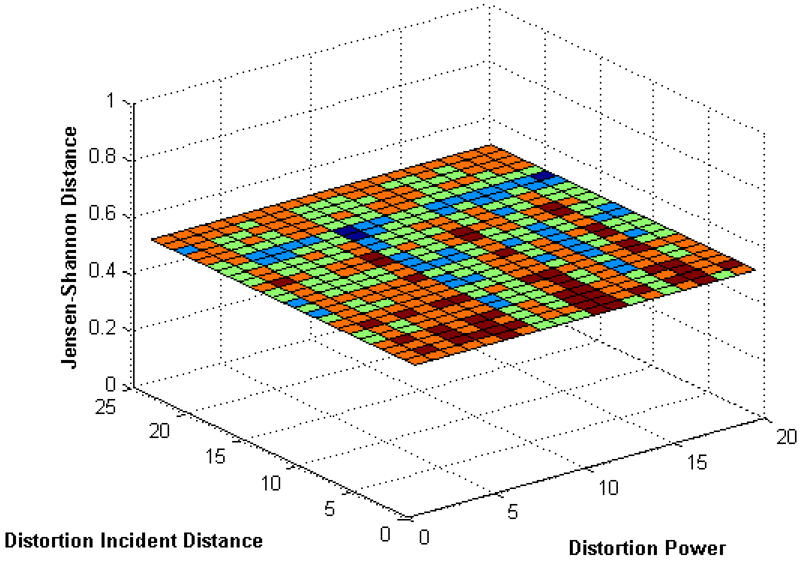


Figure 5.136: Path 2 Bias DDLHM JSds

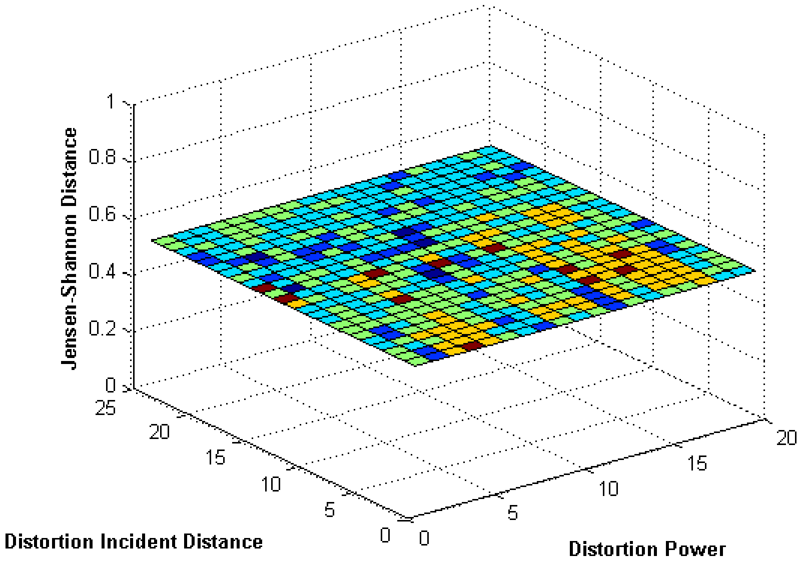


Figure 5.137: Path 2 Multipath DDLHM JSds

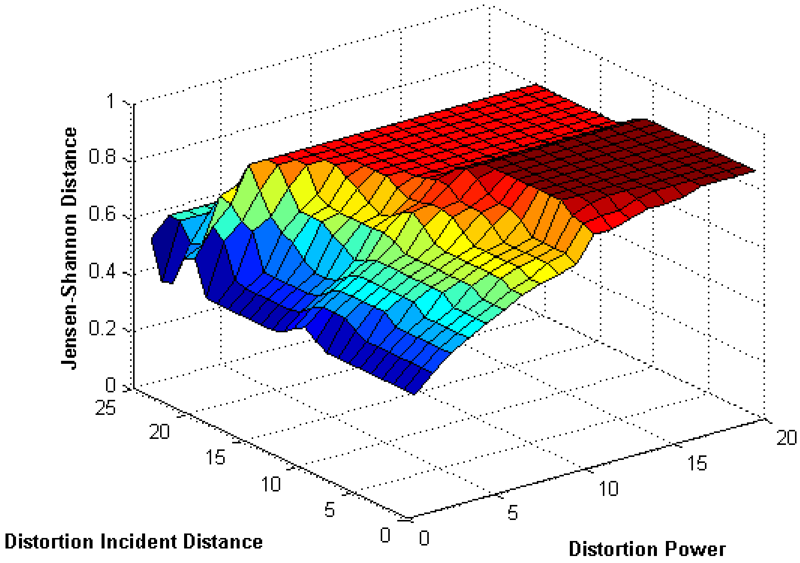


Figure 5.138: Path 3 Attenuation DDLHM JSds

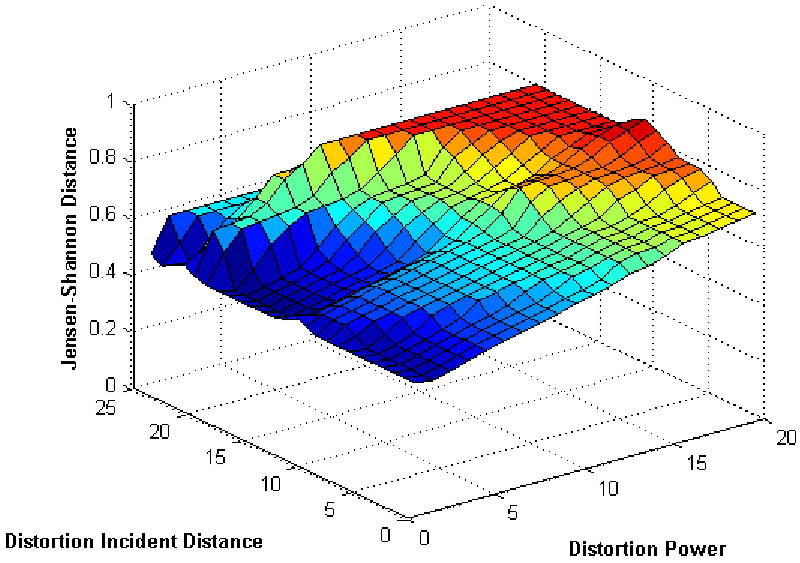


Figure 5.139: Path 3 Bias DDLHM JSds

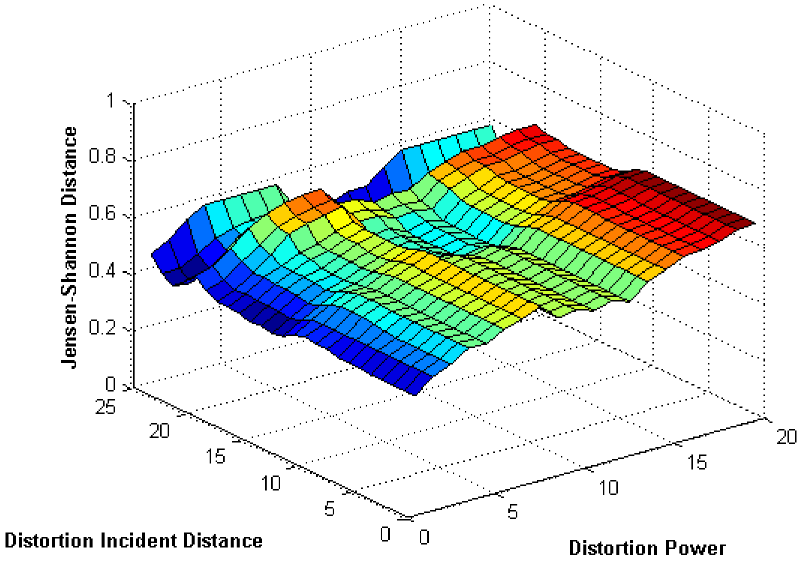


Figure 5.140: Path 3 Multipath DDLHM JSds

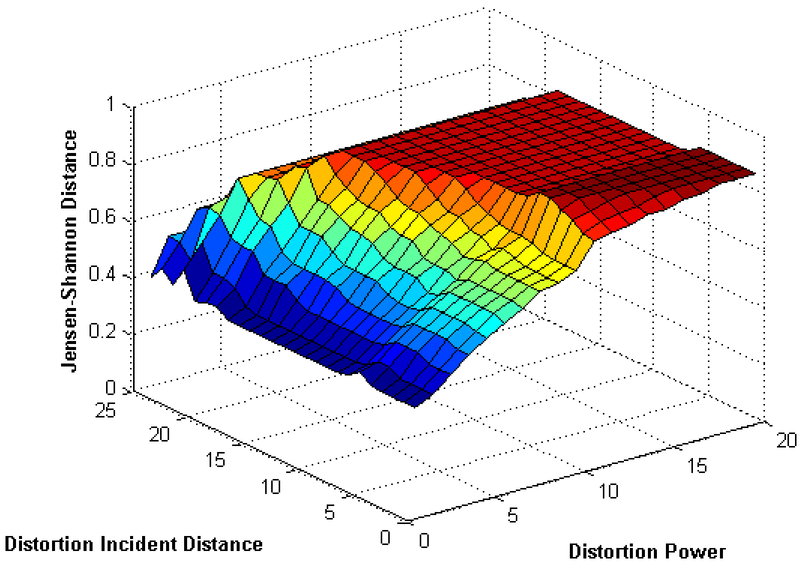


Figure 5.141: Path 4 Attenuation DDLHM JSds

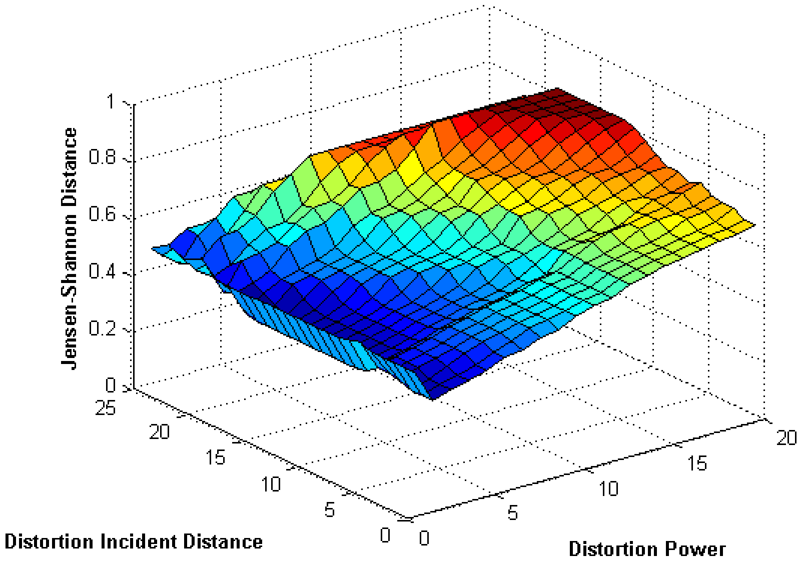


Figure 5.142: Path 4 Bias DDLHM JSds

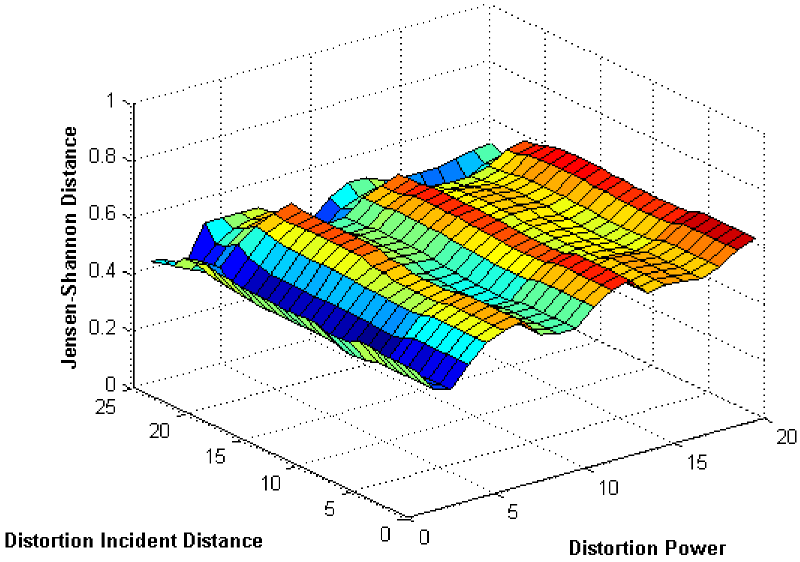


Figure 5.143: Path 4 Multipath DDLHM JSds

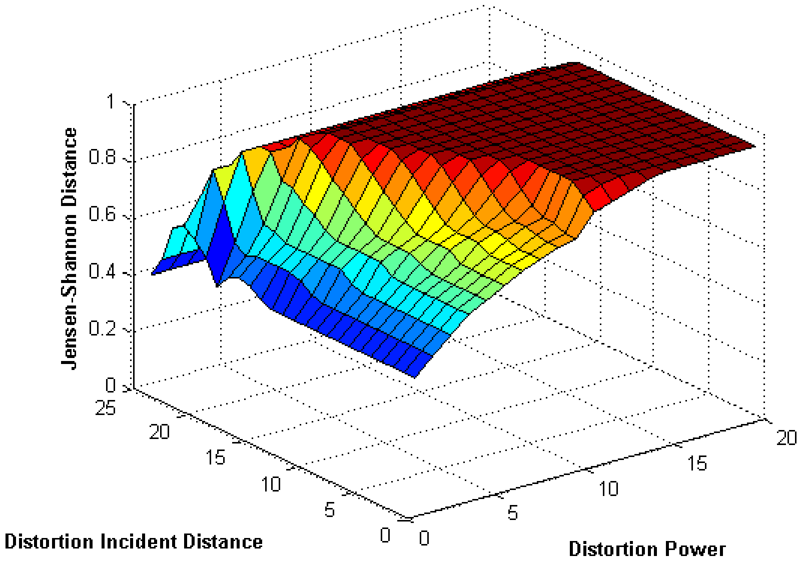


Figure 5.144: Path 5 Attenuation DDLHM JSds

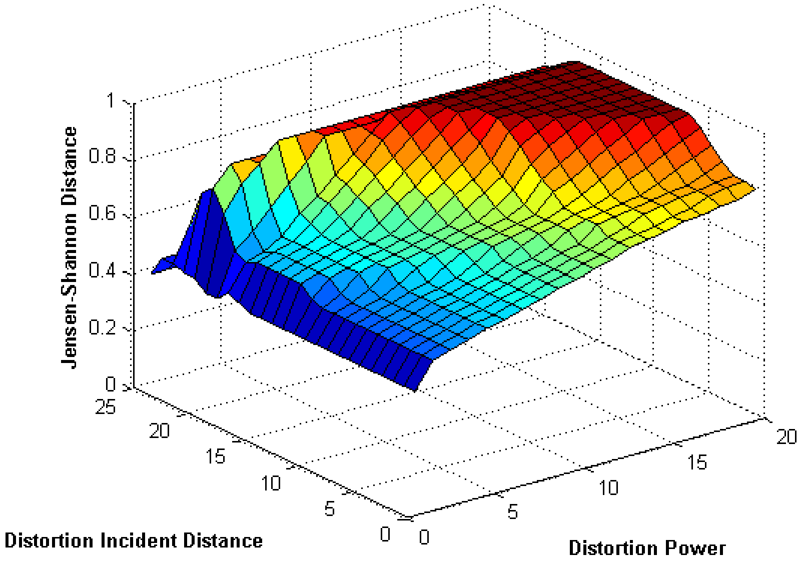


Figure 5.145: Path 5 Bias DDLHM JSds

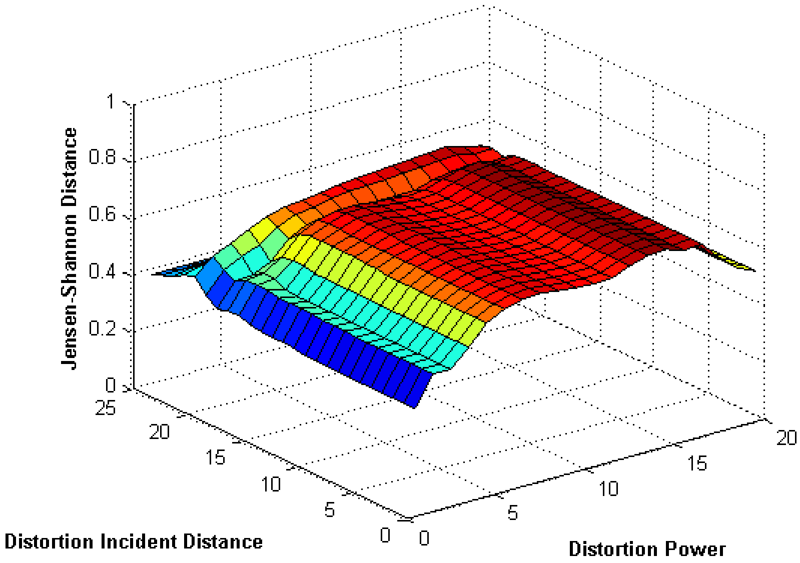


Figure 5.146: Path 5 Multipath DDLHM JSds

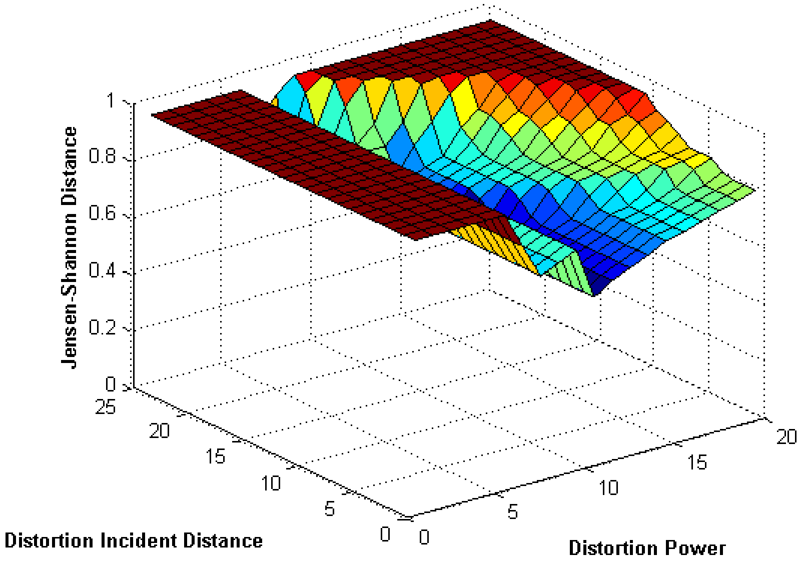


Figure 5.147: Path 6 Attenuation DDLHM JSds

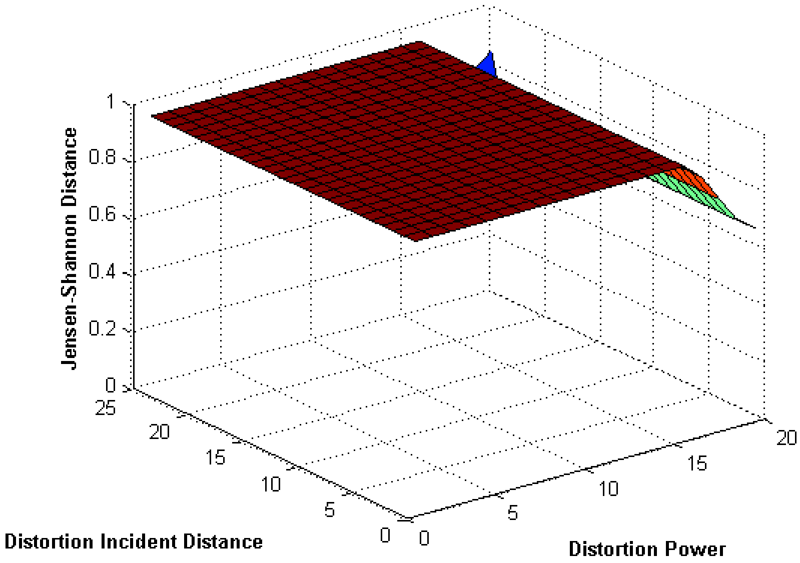


Figure 5.148: Path 6 Bias DDLHM JSds

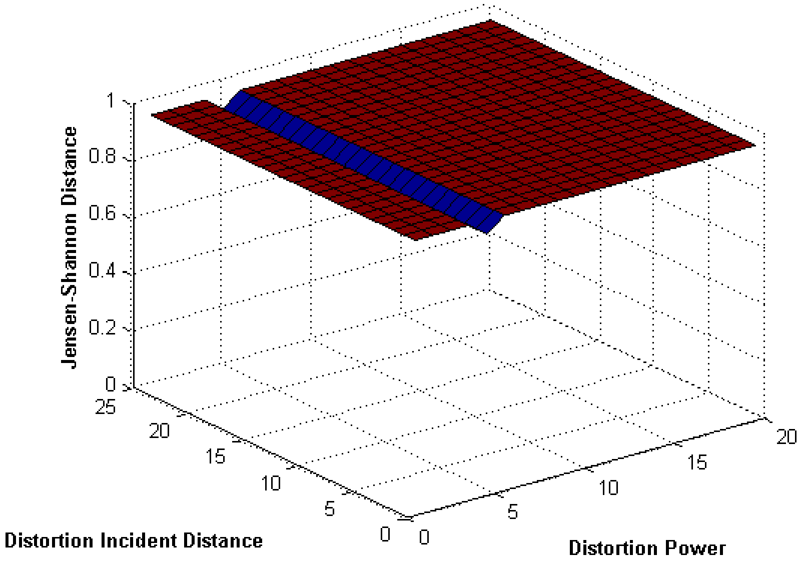


Figure 5.149: Path 6 Multipath DDLHM JSds

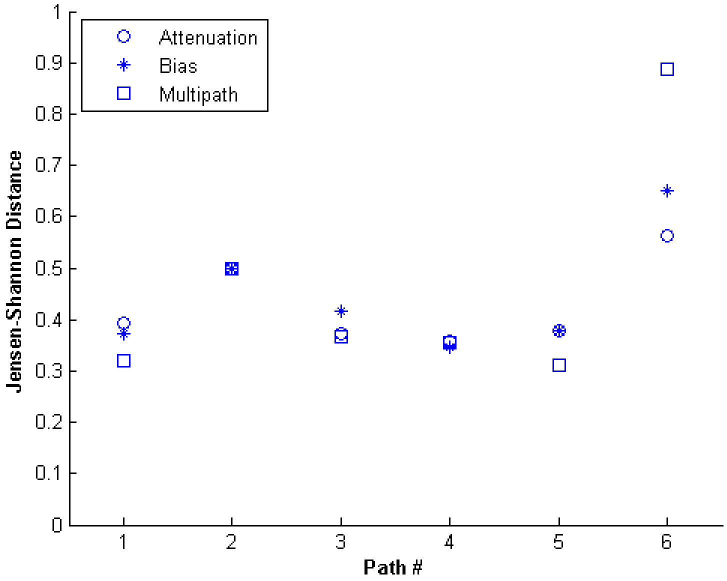


Figure 5.150: WINLAB Best Distortion Matches per Distortion

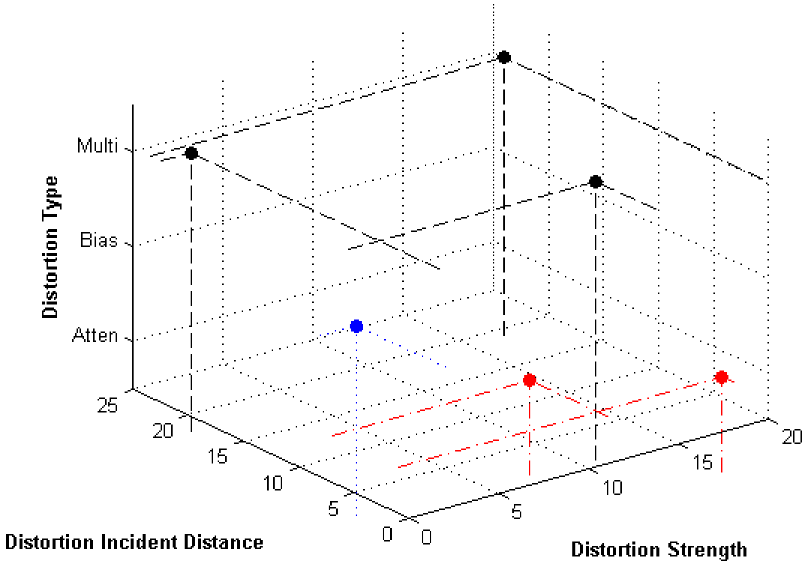


Figure 5.151: WINLAB Best Distortion Matches Overall

For each path defined in WINLAB , we computed the JSd between the DDLHM of the path's signals and the reduced parameter set. These JSds per distortion type, strength and incident distance are in Figures 5.132 through 5.149. Paths 1, 3 and 5 all prefer multipath distortion types, with a strong depression in JSd values along a segment of the multipath DDLHM distortion power range. Path 4 is not particularly diagnostic, however bias has the lowest absolute JSd. Path 2 and 6 both have fairly odd mechanics to very different ends. Path 2 has extremely similar JSd values for all distortion types and configurations, making it extremely difficult to classify. The absolute lowest JSd is a bias configuration, so we will accept it as the best possible fit given the distortion types and parameterizations considered. Path 6 is quite distinct in that it has all but no similarity to any distortion type but attenuation. It is of particular note that no distortion type or parameterization matched well, with an average minimal overall JSd of nearly 0.4 for paths 1 through 5, with Path 6's best match well above 0.5 JSd. From this we can conclude the WINLAB environment would have nearly universally inconsistent distortion characteristics, with some segments experiencing a range of medium-strength distortions and other segments fairly unobstructed, with occasional very high distortions. This behavior is expected due to the heterogeneity of the environment. Some sections are unobstructed walkways that then pass cubicles, storage cabinets and other furniture, providing both refelctive and absorptive obstructions.

5.2 Estimating Environmental Error Bounds

Some of the most significant barriers to the exact and precise development of localization algorithms are the inability to compare algorithms' performance across environments or to generate sound a-priori error expectations. Without the ability to generate a hypothesis based on a testable, identifiable cause for error any improvement on an algorithm would have to be either heuristic or a general statistical argument. While there is nothing fundamentally incorrect about non-deterministic modifications to algorithms, it can not be known if a later test's results are due to the alteration of the algorithm or a change in the test environment. Given our environmental assessments computed above in Section 5 we can determine the expected distortion types and parameters in a given environment. Using our environmental assessments in conjunction with the algorithm benchmarks we computed in Section 4.2, we can determine how much localization error would result from each parameterized distortion type for a benchmarked algorithm when it localizes along a sampled environment path. In order to effectively use these per-path error expectations to determine an average error expectation for the entire environment, we will use two main metrics and a threshold calculation to determine when to switch between them.

5.2.1 Environmental Error Expectation Metrics

As noted in paragraphs 3.4.1 through 3.4.1, the JSd between the best-matching reduced set parameterization and a series of signal samples can be interpreted as a degree of confidence that the error incurred when localizing points generated according to that parameterization will in fact mirror the error when localizing the sampled points in the actual environment. We apply this metaphor directly to compute the Weighted Environmental Error Estimate by computing the weighted average of all the paths' benchmarked errors. The weight for each benchmark error is the JSd it matched with divided by the sum of all paths' minimal JSds. This calculation strongly weights paths very high JSds, which may seem counter-intuitive since a high JSd indicates an inaccurate match. A path whose best reduced set match has a very high JSd consists of signals that can not easily be described by a single distortion type of any parameterization. If a signal path is not dominated by a single distortion, it is then the result of several strong distortions or strong noise, making it even more unlike a steady, lognormal propagation pattern and likely to cause very high localization error. Since our aim to determine how much error would likely result based on the difference between a lognormal propagation model, we find it natural to give more weight to algorithm benchmark error that is included due to weak matches over very good ones.

While the Weighted Environmental Error Estimate works well in environments that normally match reduced set parameterizations fairly well, it does not handle more noisy environments. Environments that have universally high JSd path matches the calculated weights are quite similar and the metric approaches a simple arithmetic mean. In such environments the collected benchmark errors need to be scaled to an appropriate amount to reflect the fact that they come from imprecise matches, resulting in the Scaled Environmental Error Estimate. We first compute the Weighted Environmental Error Estimate and subtract from it the mean of all the environment's paths' benchmark errors. We then divide the difference of the maximal and minimal benchmark error by this quantity to produce an error scale and multiply the mean of the benchmark errors by it.

In order to choose between the Weighted and Scaled Environmental Error Estimates, we examine the mean of the JSds of all path matches. If the mean is relatively low, below 0.1, we presume that there are relatively few high-JSd paths, and their error will be appropriately diagnostic, and employ the Weighted Environmental Error Estimate. If the mean is relatively high, above 0.15, we presume there are many high-JSd paths and that their error will not be particularly diagnostic, and instead employ the Scaled Environmental Error Estimate. In all cases we will compare our estimated environmental error to the average localization error when localizing all points in each live environment. Since most of the test environments do not consist of only straightline paths, we will be using a subset of the total environment to draw conclusions about localization performance over the rest of it.

5.2.2 CoRE Expected Error

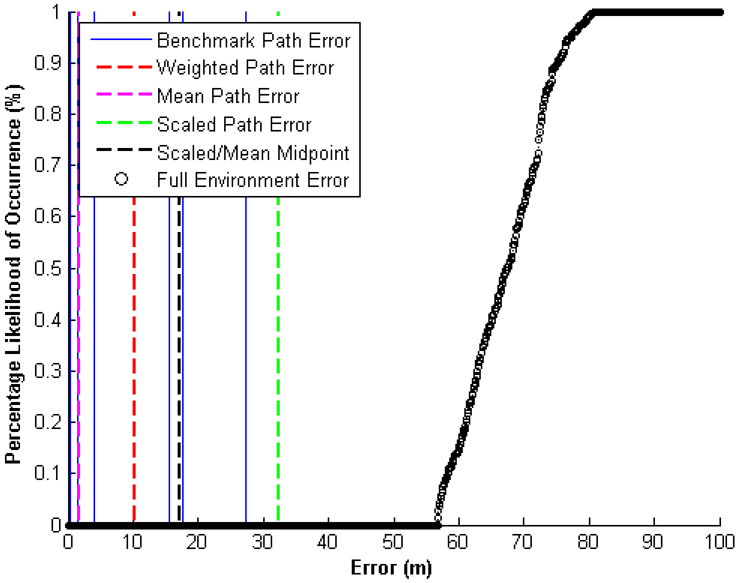


Figure 5.152: Algorithm Error in CoRE: RT Error and Expectations

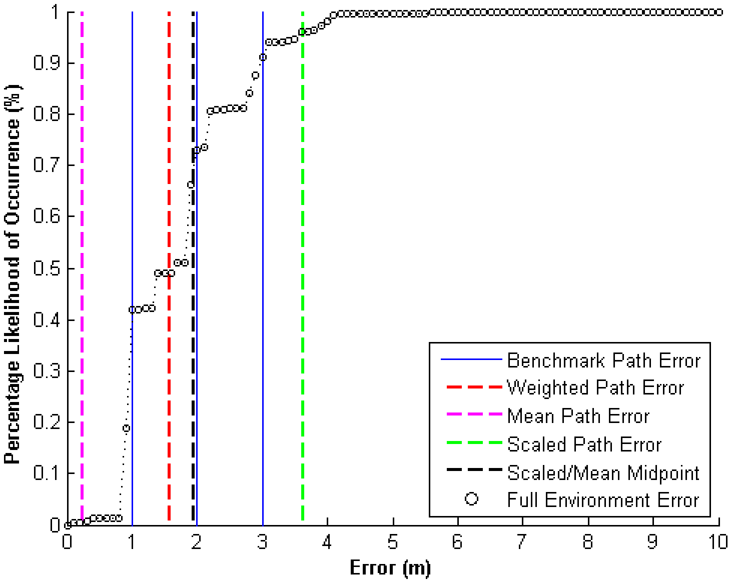


Figure 5.153: Algorithm Error in CoRE: RADAR Error and Expectations

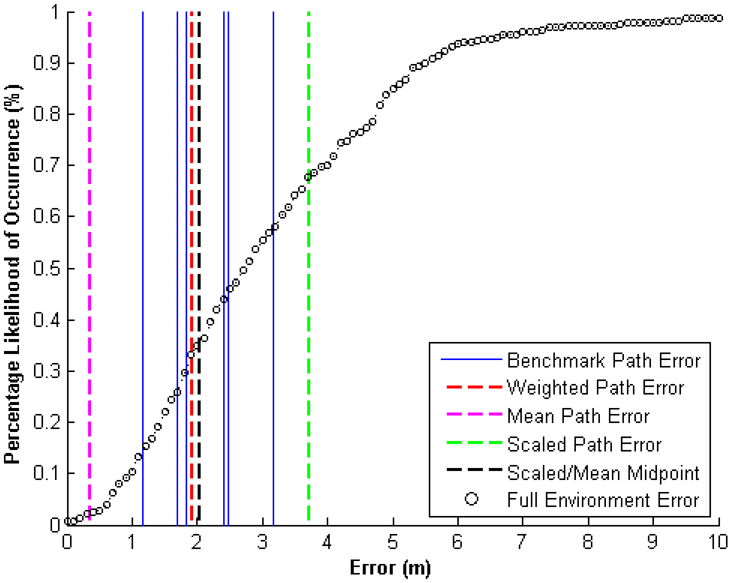


Figure 5.154: Algorithm Error in CoRE: ABP Error and Expectations

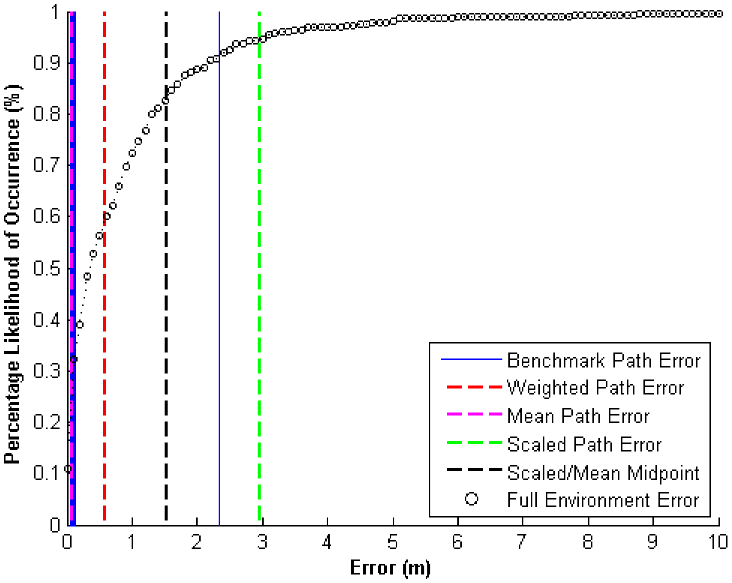


Figure 5.155: Algorithm Error in CoRE: SPM Error and Expectations

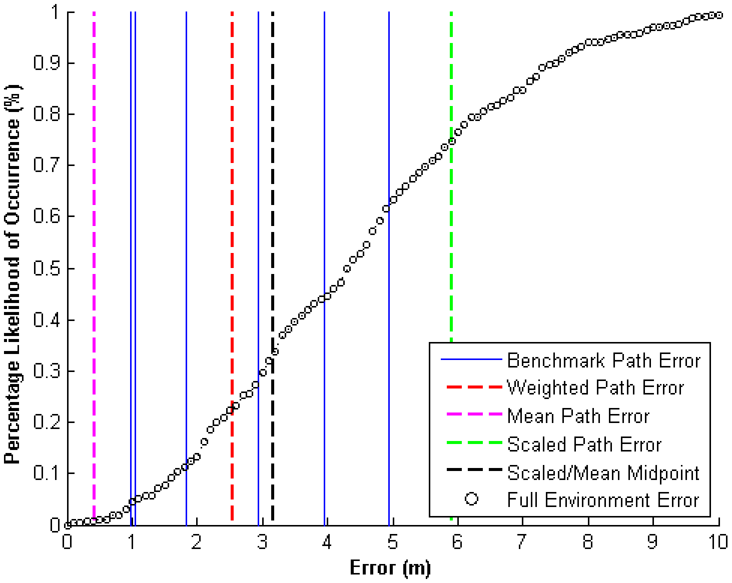


Figure 5.156: Algorithm Error in CoRE: M1 Error and Expectations

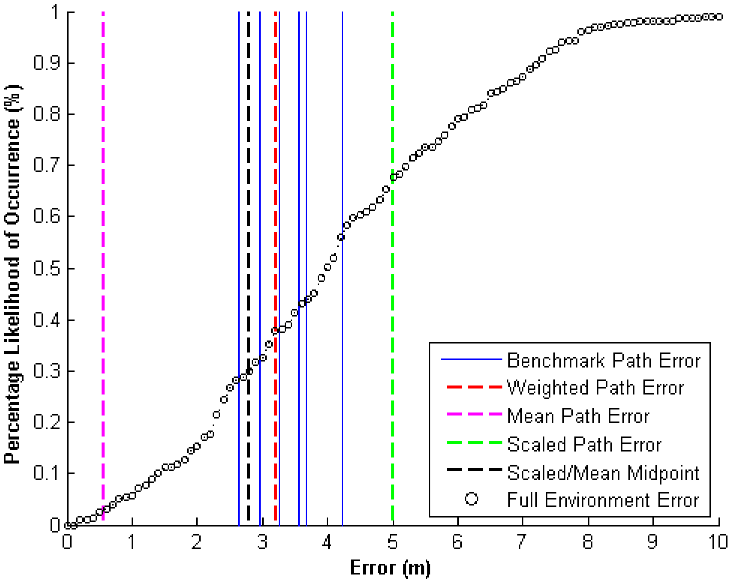


Figure 5.157: Algorithm Error in CoRE: M2 Error and Expectations

	Rotated Trilateration	RADAR	ABP	SPM	M1	M2
Error Estimate	10.24	1.57	1.92	0.59	2.55	3.21
Actual Mean Error	67.4	1.4	2.7	0.3	4.3	4
Difference	57.16	0.17	0.78	0.29	1.75	0.79

Table 5.1: Estimated and Actual Mean Error for Core

In Figures 5.152 through 5.157, we have graphed the individual benchmark error as well as the error CDF that results from localizing across the entire environment, including points that our paths did not cover. We also include both our Weighted and Scaled Environmental Error Estimates as well as some diagnostic additional measurements to better determine the context of the measurements. The mean of the JSds for all Core path matches is 0.0801, indicating that the Weighted Environmental Error Estimate should be used. As can be seen in Table 5.1, all algorithms but Rotated Trilateration have fairly solid estimates within 2 meters of their actual values, and of those only M1's estimate falls higher than 1 meter away from the overall environmental average error. Since Rotated Trilateration is engineered to have very strong error responses, even a very small deviation in parameter classification can cause a very large misprediction of error. These prediction results are consistent with our expectations and analysis of the Core distortion characteristics in paragraph 5.1.2 and the relatively low average match JSd. The relatively low average match JSd of 0.0801 indicates that many of the paths analyzed match quite well to a particular parameterization. The standard deviation of distribution of match JSds is however 0.1974, indicating that the match JSds are not definitely not uniformly low, but that there are a few matches with particularly high JSds, which is borne out by the environmental path analysis in Subsection 5.1.2 above.

5.2.3 Grid Expected Error

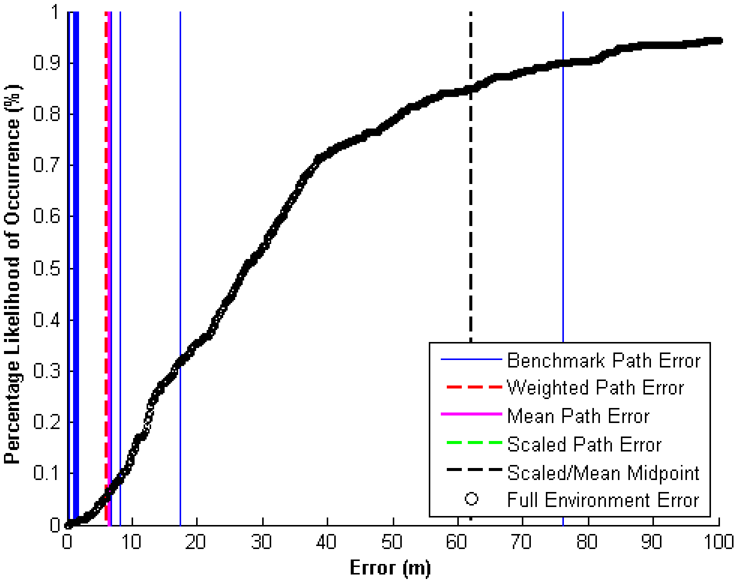


Figure 5.158: Algorithm Error in Grid: RT Error and Expectations

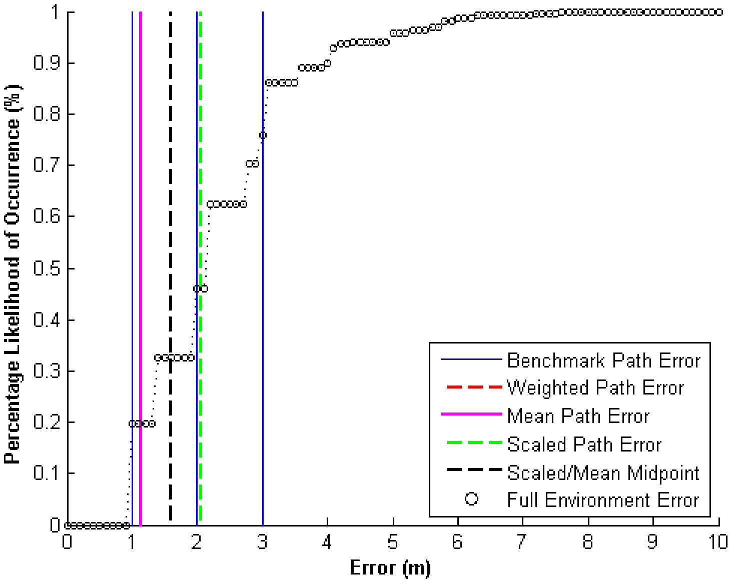


Figure 5.159: Algorithm Error in Grid: RADAR Error and Expectations

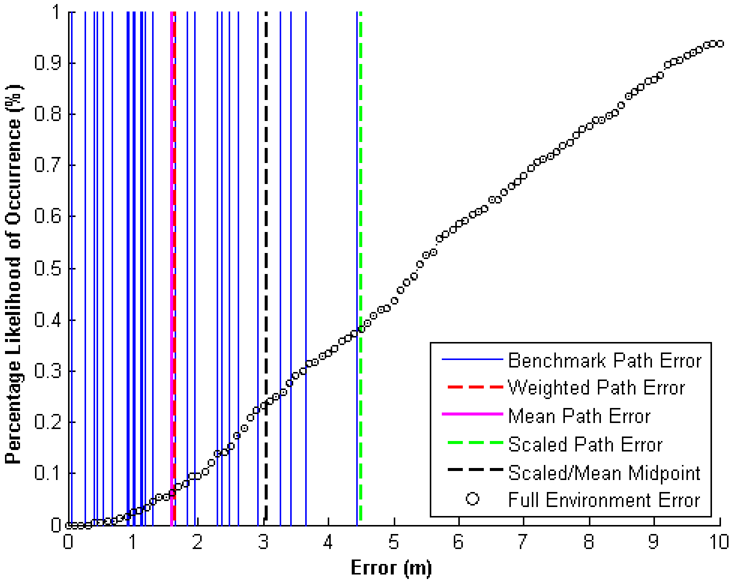


Figure 5.160: Algorithm Error in Grid: ABP Error and Expectations

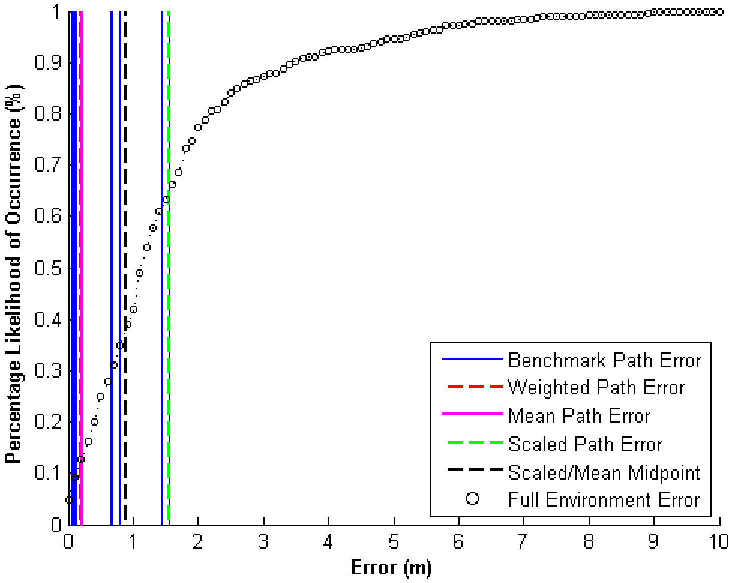


Figure 5.161: Algorithm Error in Grid: SPM Error and Expectations

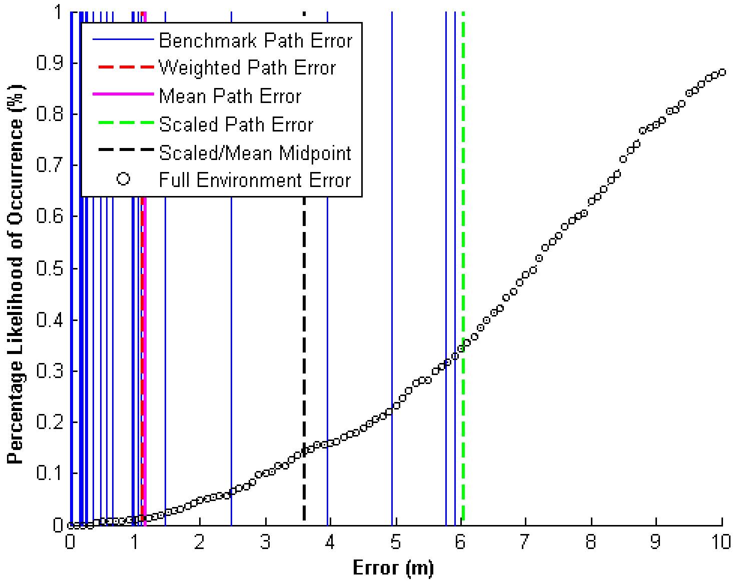


Figure 5.162: Algorithm Error in Grid: M1 Error and Expectations

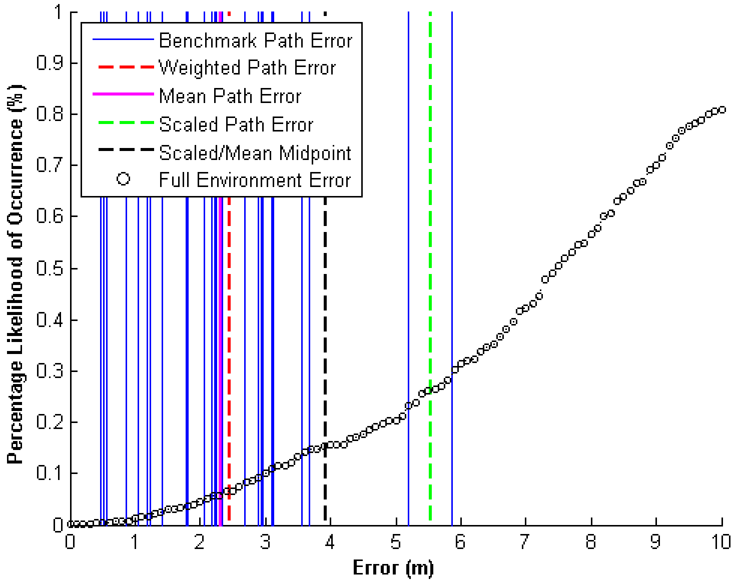


Figure 5.163: Algorithm Error in Grid: M2 Error and Expectations

	Rotated Trilateration	RADAR	ABP	SPM	M1	M2
Error Estimate	117.83	2.06	4.51	1.55	6.05	5.54
Actual Mean Error	27.5	2	5.4	1.1	7.1	7.5
Difference	90.33	0.06	0.89	0.45	1.05	1.96

Table 5.2: Estimated and Actual Mean Error for Grid

In Figures 5.158 through 5.163, we have graphed the individual benchmark error as well as the error CDF that results from localizing across the Grid environment. The Grid environment is distinct from others in that it consists of a single, open room with occasional support beams as the only obstructions. The room however represents a significant localization challenge as many of the surfaces are strongly reflective; the floor is tiled, the ceiling made of a thin, corrugated metal and one wall consists of mostly plate glass windows. All these materials can be highly radioreflective at the correct incident angle. Given that nearly all points in the room have an unobstructed path to these environmental features, it is quite likely that the signals sampled in the environment would universally experience a high degree of distortion in comparison to other environments. The mean of the JSds for all Grid path matches is 0.2851, indicating that the Scaled Environmental Error Estimate should be used. As can be seen in Table 5.2, all algorithms but Rotated Trilateration have fairly solid error estimates within 2 meters of the overall average error across the entire environment, and of these only M2's estimate exceeds 1 meter by an appreciable amount. We find these results quite encouraging in the face of the Grid environment's fairly high match JSd average of 0.2851, indicating a large number of fairly imprecise matches. The standard deviation of the distribution of JSd matches is 0.0639, indicating the relatively high mean is fairly stable across all matches. This is fairly deleterious as a JSd of 0.2851 would lie outside the range of a 'good match' as per the criteria we established in Section 3.4. Even so, our estimation method has proven to be quite robust even in such an environment for all algorithms but Rotated Trilateration, which we engineered to be specifically incredibly sensitive to deviations from unobstructed lognormal. Given the high average JSd and resultant misparameterizations, it is reasonable to presume Rotated Trilateration's performance would be difficult to estimate without a much more precise distortion assessment tool and benchmarks.

5.2.4 WINLAB Expected Error

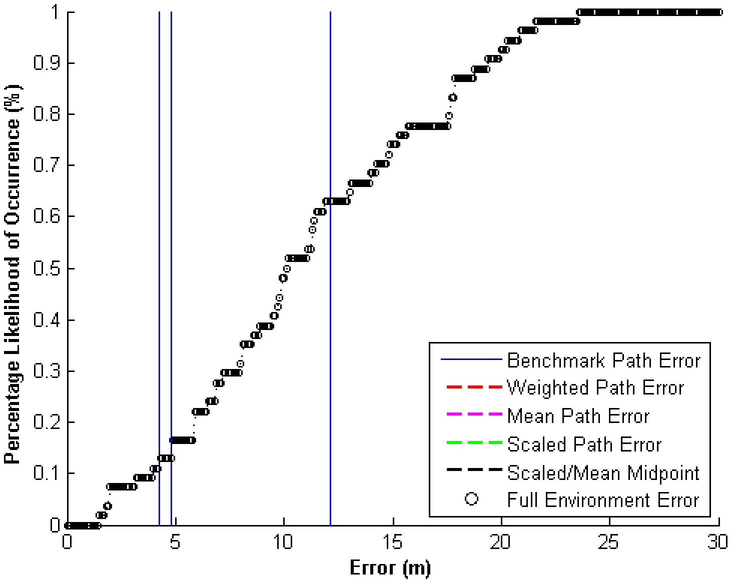


Figure 5.164: Algorithm Error in WINLAB: RT Error and Expectations

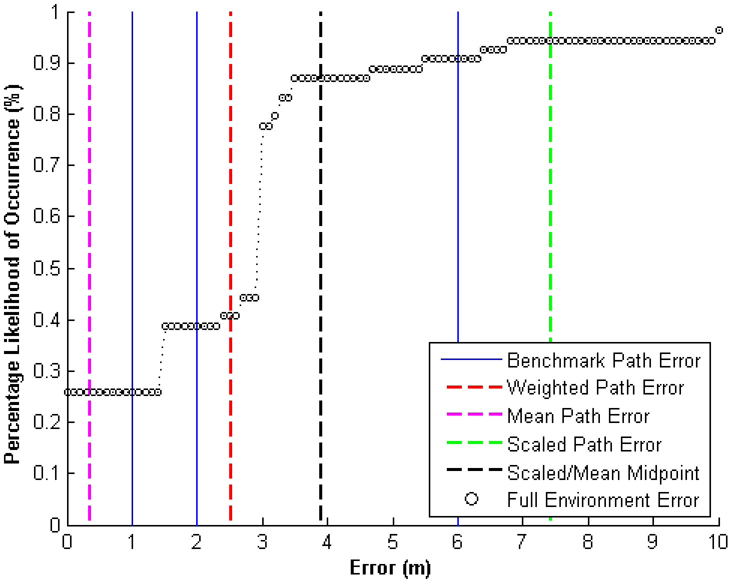


Figure 5.165: Algorithm Error in WINLAB: RADAR Error and Expectations

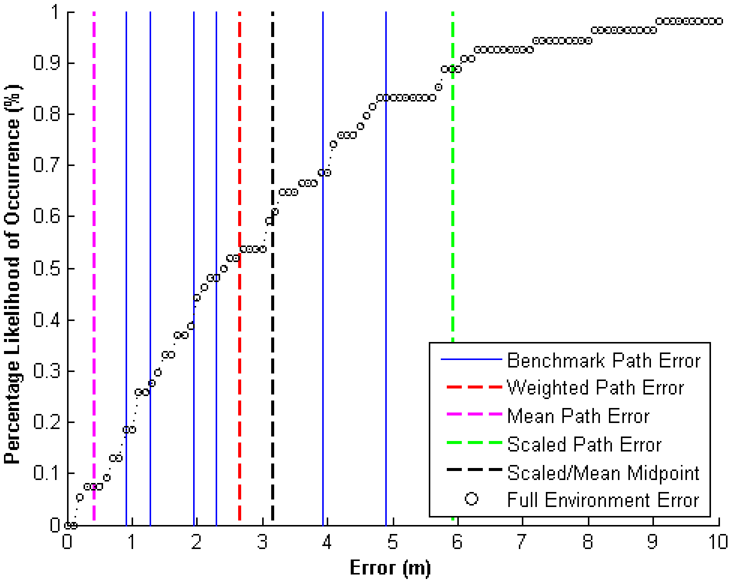


Figure 5.166: Algorithm Error in WINLAB: ABP Error and Expectations

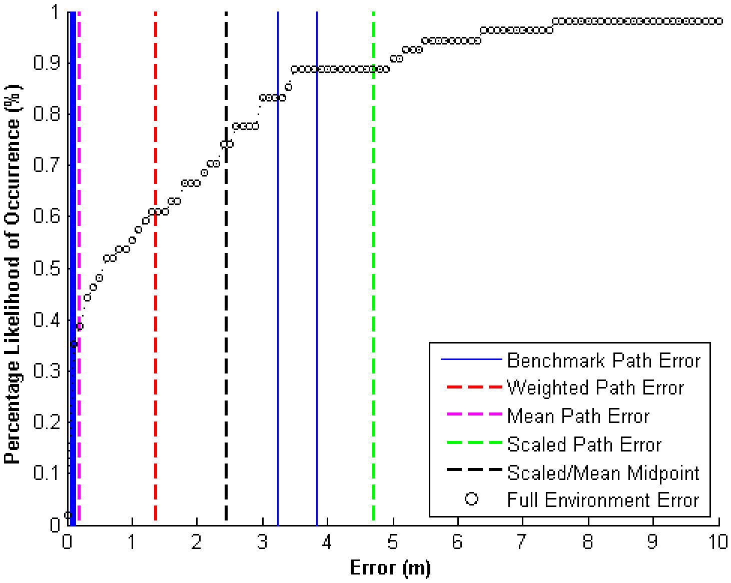


Figure 5.167: Algorithm Error in WINLAB: SPM Error and Expectations

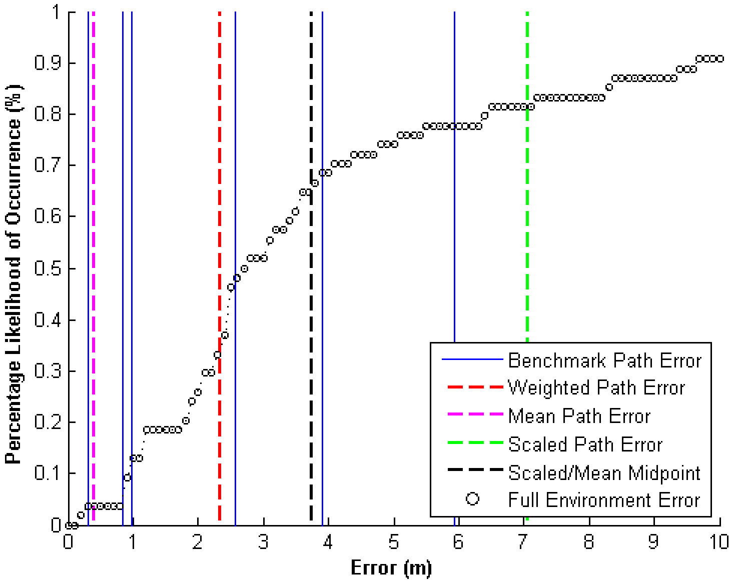


Figure 5.168: Algorithm Error in WINLAB: M1 Error and Expectations

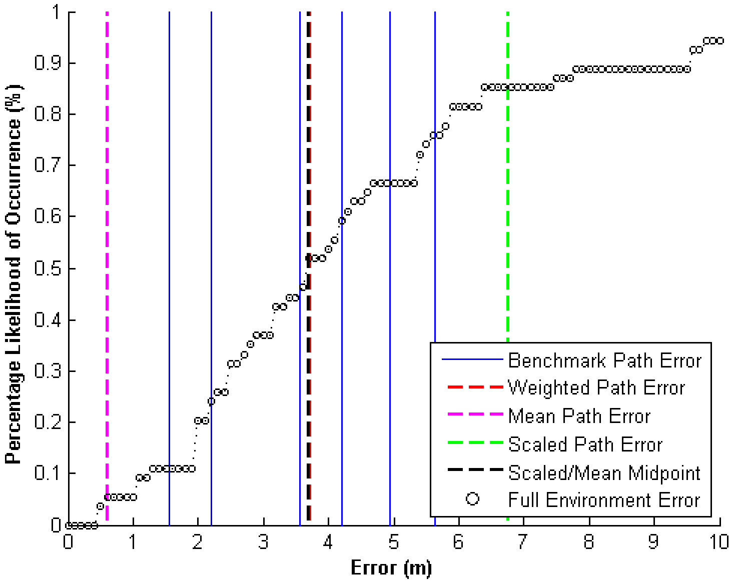


Figure 5.169: Algorithm Error in WINLAB: M2 Error and Expectations

	Rotated Trilateration	RADAR	ABP	SPM	M1	M2
Error Estimate	42467774.87	2.52	2.67	1.36	2.33	3.72
Actual Mean Error	10.1	2.7	2.4	0.6	2.7	3.7
Difference	42467764.87	0.18	0.27	0.76	0.37	0.02

Table 5.3: Estimated and Actual Mean Error for WINLAB

In Figures 5.164 through 5.169, we have graphed the individual benchmark error as well as the error CDF that results from localizing across the WINLAB environment. The WINLAB environment is distinct from others in that it consists of a mix of hallways and an room populated with desks and half-height cubicle walls. The hallways resemble the Grid environment, as they are tiled and have a corrugated metal ceiling. The cubicle area however is carpeted, but also has glass-fronted offices. Even given the general similarities of the WINLAB environment to the Grid environment, we expect much less distortion from it. In the cubicle area the carpeted floor, cubicle walls and desks will likely handily absorb reflections. In the hallway areas the closer walls and support materials are likely absorb, attenuate or reflect away signals, keeping non-line-of-sight signals from propagating down the hallway and likely reducing the amount of noise or signal churn in the environment. The mean of the JSds for all WINLAB path matches is 0.067, indicating that the Weighted Environmental Error Estimate shoudl be used. As can be seen in Table 5.3, once again all algorithms but Rotated Trilateration have very good error estimates. All are within half a meter of the overall average across all points for their respective algorithm save for SPM and Rotated Trilateration, with SPM's estimated error average still falling under a meter of the actual. Rotated Trilateration's extraordinarily terrible performance can be wholly attributed to a single path whose match was very bad. The bad match along with Rotated Trilateration's extreme error sensitivity resulted in an error estimation incorrect by nearly 42.5 million meters. This was the product of a rather short path that had a fair amount of noise on it, resulting in a large amount of missing data, namely Path 2, whose DDLHM JSds are detailed in Figures 5.135 through 5.137. These extremely flat JSd ranges demonstrate how little information was available in the samples collected, as they barely matched to any of the parameterizations considered. The standard deviation of the distribution of all the path JSd matches is 0.1569, indicating that there are likely only one or two path estimates that are fairly bad, while the others are mostly good. This is in direct correspondence with our other data and the estimation performance. Even given the single bad estimate, we find our method to be particularly capable, as it estimates the overall average error of all algorithms, save one, to less than a meter of their actual values.

Chapter 6
Conclusions

As wireless networks continue their inexorable spread into full ubiquity they enable a host of computing applications powered by increasingly powerful, small, efficient computers and sensors. Given the proliferation of personal communication and computation devices we are presented with a degree of mobile computing undreamt of when many of the computing and communication systems and standards we rely on daily were drafted. We are presented with the distinctly difficult task of building up new systems out of pieces not meant to support such operations or organizations. Location is often a lynchpin that holds many context-based and sensor applications together. Other than time and physical state, few sensing systems do not use location in some manner. Given the wide deployment of wireless networks it is particularly compelling to reuse our indoor communication networks as location sensing systems as well, to immediately add a location context to any device we can communicate with absolutely no additional software, hardware, cooperation or collusion on the part of the device by using passive signal sensing. Without a measuring or compensating mechanism, it is not possible to attribute localization error to the algorithm or the environment, making rational analysis and precise improvements infeasible. Developing localization algorithms in the quantitative vacuum of quantum probabilities and gestalt results halts any such progress, leaving us in the realm of heuristic and analogy.

By rigorously applying the philosophical precepts of the metaphysic of artificial knowledge set forward in Immanuel Kant's *Critique on Pure Reason*, demonstrated how to: benchmark localization algorithms' performance in the presence of precise amounts of distortion, detect environmental distortion and match it to our distortion scenarios, and then use these two processes to generate extremely accurate error predictions for localization algorithms computing in a given environment. Beyond prediction, our algorithmic benchmarking methods provide a tool to assess an algorithm's error response in order to inform development and to diagnose performance issues. Our environmental assessment methods allow environments to be compared quantitatively and for localization algorithms' performance to be understood and analyzed in a common context.

6.1 Future Work

While our synthetic error model does a good job of estimating environmental error, it does so by considering only one of three dominant radio distortion types. Our method could be improved by iterative reapproximation, estimating the strongest distortion type, removing its major components, and then estimating again to gain an arbitrary degree of improvement, although each additional layer of estimation requires exponentially more parameterizations to be considered, making a pruning mechanism essential.

The real potential of this process is to generalize it beyond localization. There is no inherent part of the process that ties it to localization, but the assumption of the distortion types. By replacing the distortion type models with a typicality analysis and a curve fitting step, distortion models should be extractable from the object data itself. The most typical sequence should mirror the presumed model, so, in the case of localization, the best-fitting lognormal values would be subtracted from a series of signal paths and the paths analyzed in order to isolate the next most typical sequence. In this manner a series of distortions models could be directly estimated from the data. Since a priori knowledge of the error types is not required, our processes could be potentially applied to any data, model and environment, whether it be exact, estimated, physical, mathematical or simulated. Currently, so long as the expected distortion types are known, our process can be applied to any model or environment to construct a synthetic error model to describe a metaphysical relation between input and output when the direct physical relation can not be known.

References

[1] A. Banerjee, D. Maas, M. Bocca, N. Patwari, S. Kasera. Violating privacy through walls by passive monitoring of radio windows. In *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks*, WiSec, pages 69–80. ACM, 2014.

[2] A. Borrelli, C. Monti, M. Vari, F. Mazzenga. Channel models for ieee 802.11b indoor system design. In *IEEE International Conference on Communications*, pages 3701–3705, June 2004.

[3] A. Neskovic, N. Neskovic, G. Paunovic. Modern approaches in modeling of mobile radio systems propagation environment. *IEEE Communications Surveys Tutorials*, pages 2–12, May 2000.

[4] Lauri Aalto, Nicklas Göthlin, Jani Korhonen, and Timo Ojala. Bluetooth and wap push based location-aware mobile advertising system. In *International Conference on Mobile Systems, Applications, and Services (MobiSys)*, pages 49–58. ACM, 2004.

[5] K. I. Ahmed and G. Heidari-Bateni. Improving two-way ranging precision with phase-offset measurements. In *Global Telecommunications Conference (GLOBECOM)*, pages 1–6, November 2006.

[6] A.N. D'Andrea, U. Mengali, R. Reggiannini . The modified cramer-rao bound and its application to synchronization problems. *Communications, IEEE Transactions on*, 42(234):1391–1399, Feb 1994.

[7] M.I. Husain A.T. Parameswaran and S. Upadhyaya. Is rssi a reliable parameter in sensor localization algorithms - an experimental study. Technical report, State University of New York at Buffalo, 2009.

[8] R. L. Kashyap B. Yazici. Signal modeling and parameter estimation for 1/f processes using scale stationary models. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 2851–2844, May 1996.

[9] P. Bahl and V. N. Padmanabhan. Radar: An in-building rf-based user location and tracking system. In *IEEE Conference on Computer Communications (INFOCOMM)*, 2000.

[10] Sangwook Bak, Seokseong Jeon, Young-Joo Suh, Chansu Yu, and Dongsoo Han. Characteristics of a large-scale wifi radiomap and their implications in indoor localization. In *International Conference on the Network of the Future (NOF)*, pages 1–5, Oct 2013.

[11] Biaz S, Ji Y, Qi B, Wu S. Realistic radio range irregularity model and its impact on localization for wireless sensor networks. *International Conference on Wireless Communications, Networking and Mobile Computing (WCNMC)*, 2005.

[12] P. Castro, P. Chiu, T. Kremenek, and R. R. Munz. A probalistic room location service for wireless networked environments. In *In Proceedings of the 3rd Annual Conference on Ubiquitous Computing (Ubicomp)*, 2001.

[13] Yingying Chen, J. Francisco, W. Trappe, and R. Martin. A practical approach to landmark deployment for indoor localization. In *Sensor and Ad Hoc Communications and Networks (SECON)*, volume 1, pages 365–373, Sept 2006.

[14] D. Joho, C. Plagemann, W. Burgard. Modeling rfid signal strength and tag detection for localization and mapping. In *International Conference on Robotics and Automation*, pages 3160–3165, May 2009.

[15] D. Zhang, K. Lu, R. Mao, Y. Feng, Y. Liu, Z. Ming and L.M. Ni. Fine-grained localization for multiple transceiver-free objects by using rf-based technologies. *Parallel and Distributed Systems, IEEE Transactions on*, 25(6):1464–1475, June 2014.

[16] Eiman Elnahrawy, Xiaoyan Li, and Richard P. Martin. The limits of localization using signal strength: A comparative study. In *Proceedings of teh 1st Annual IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON)*, 2004.

[17] M. Vari F. Mazzenga F. Capulli, C. Monti. Path loss models for ieee 802.11b wireless local area networks. *IEEE International Conference on Communications Systems (ISWCS)*, 2004.

[18] M. Vari F. Mazzenga F. Capulli, C. Monti. Path loss models for ieee 802.11a wireless local area networks. *International Symposium on Wireless Communication Systems (ISWCS)*, 2006.

[19] Andreas Haeberlen, Algis Rudys, Eliot Flannery, Dan S. Wallach, Andrew M. Ladd, and Lydia E. Kavraki. Practical robust localization over large-scale 802.11 wireless networks. In *International Conference on Mobile Computing and Networking (MOBICOM*, pages 70–84. ACM Press, 2004.

[20] J. Fink, N. Michael, A. Kushleyev, V. Kumar. Experimental characaterization of radio signal propagation in indoor environments with application to estimation and control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2834–2839, October 2009.

[21] M. Haenggi K. Woyach, D. Puccinelli. Sensorless sensing in wireless networks: Implementation and measurements. In *International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pages 1–8, April 2006.

[22] A. Napolitano M. Ficco, C. Esposito. Calibrating indoor positioning systems with low efforts. *IEEE Transactions on Mobile Computing*, 2014.

[23] Y. Ohta M. Murata M. Sugano, T. Kawazoe. Indoor localization system using RSSI measurement of wireless sensor network based on zigbee standard. In *IASTED International Multi-Conference on Wireless and Optical Communications*, pages 3–5, July 2006.

[24] David Madigan, Eiman Elnahrawy, Richard P. Martin, Wen-Hua Ju, P. Krishnan, and A. S. Krishnakumar. Bayesian indoor positioning systems. In *IEEE Conference on Computer Communications (INFOCOMM)*, pages 1217–1227, 2005.

[25] V. Moghtadaiee and A.G. Dempster. Wifi fingerprinting signal strength error modeling for short distances. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–6, Nov 2012.

[26] Veljo Otsason, Alex Varshavsky, Anthony Lamarca, and Eyal De Lara. Accurate gsm indoor localization. In *In the proc. of UbiComp 2005*, pages 141–158, 2005.

[27] N. Patwari and P. Agrawal. Effects of correlated shadowing: Connectivity, localization, and rf tomography. In *International Conference on Information Processing in Sensor Networks*, IPSN, pages 82–93, April 2008.

[28] Neal Patwari, Alfred O. Hero III, Alfred O. Hero, Matt Perkins, Robert J. O'Dea, Neiyer S. Correal, and Robert J. O'dea. Relative location estimation in wireless sensor networks. *IEEE Transactions on Signal Processing*, 51:2137–2148, 2003.

[29] R. Peng and M. Sichitiu. Angle of arrival localization for wireless sensor networks. In *Sensor and Ad Hoc Communications and Networks (SECON)*, 2006.

[30] Nissanka Bodhi Priyantha, Hari Balakrishnan, and Bodhi Priyantha. The cricket indoor location system. Technical report, Massachusetts Institute of Technology, 2005.

[31] S. Mazuelas, A. Bahillo, R.M. Lorenzo, P. Fernandez, F.A. Lago, E. Garcia, J. Blas, and E.J. Abril. Robust indoor positioning provided by real-time rssi values in unmodified wlan networks. *IEEE Journal of Selected Topics in Signal Processing*, 3(5):821–831, Oct 2009.

[32] M. Sakurada and M. Fukuda. An rssi-based error correction applied to estimated sensor locations. In *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, pages 58–63, Aug 2013.

[33] R. S. Saunders. *Antennas and Propagation for Wireless Communication Systems*. Ed. Wiley, 1999.

[34] F. Seco, A.R. Jimenez, C. Prieto, J. Roa, and K. Koutsou. A survey of mathematical methods for indoor localization. In *Intelligent Signal Processing, 2009. WISP 2009. IEEE International Symposium on*, pages 9–14, Aug 2009.

[35] Wooju Kim Sharly Joana Halder, Joon-Goo Park. Adaptive filtering for indoor localization using zigbee rssi and lqi measurement. *Adaptive Filtering Applications*, 2011.

[36] Apostolos Traganitis Stefanos Papadakis. Wireless positioning using the signal strength difference on arrival. *IEEE 7th International Conference on Mobile Adhoc and Sensor Systems (MASS)*, 2010.

[37] Roy Want, Veronica Falcao, and Jon Gibbons. The active badge location system. *ACM Transactions on Information Systems*, 10:91–102, 1992.

[38] Andy Ward, Alan Jones, and Andy Hopper. A new location technique for the active office. In *IEEE Personal Communications*, pages 42–47, 1997.

[39] IEEE 802.11 working group. Iso/iec standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications (includes ieee std 802.11, 1999 edition; ieee std 802.11a.-1999; ieee std 802.11b.-1999; ieee std 802.11b.-1999/cor 1-2001; and ieee std 802.11d.-2001). *ISO/IEC 8802-11 IEEE Std 802.11 Second edition 2005-08-01 ISO/IEC 8802 11:2005(E) IEEE Std 802.11i-2003 Edition*, pages 1–721, 2005.

[40] H. Yucel, A. Yazici, and R. Edizkan. A survey of indoor localization systems. In *Signal Processing and Communications Applications Conference (SIU)*, pages 1267–1270, April 2014.

[41] Z. Xue, C. Tepedelenlioglu, M. Banavar, A. Spanias. Crlb for the localization error in the presence of fading. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 5150–5154, May 2013.