CALIFORNIA INSTITUTE OF TECHNOLOGY
JET PROPULSION LABORATORY

New Technology Reporting Form **NTR Number: 44810**

Title: New Constellations for Communications Signalling: Design Methodolgy and Method and Apparatus for the New Signalling Scheme

New Technology Description

Novelty - Describe what is new and different about your work. Attach supporting material if necessary.

We propose new constellations to be used with capacity approaching codes such as LDPC or Turbo Codes. The new constellations are unequally spaced, and are designed to target a specific user data rate in contrast to traditional constellations. The new constellations are designed to maximize either the joint capacity or the parallel decoding capacity at a target user data rate. These constallations achieve a smaller gap to the ultimate Shannon limit. Simulations results have shown gains of more than 1dB compared to traditional constellations.

Technical Disclosure

Problem - Motivation that led to development or problem that was solved.

The introduction of Turbo Codes and LDPC codes in the nineties allowed for very powerful coding schemes achieving near Shannon capacity performance for BPSK/ QPSK. However, the gap to capacity increases with bandwidth efficiency (ie as more bits are packed per transmitted symbol) when using traditional constellations with modern codes. While the modern codes are highly optimized, the traditional constellations aren't.

Solution

We developed new constellations that are:
1) unequally spaced
2) jointly optimized for location and bit labels
3) designed to target a specific user data rate
4) are designed to maximize either the joint capacity or parallel decoding capacity.
5) achieve a smaller gap to the Shannon limit

Description

As per the attached document, using numerical optimization we designed PAM constellations for different bandwidth efficiencies and different user data rates. We also give a sample optimized PSK constellation. Tables of these constellations can be provided upon request for a patent application or any other purpose. QAM constellations can be constructed using the optimized PAM constellations.

Constellation Design via Capacity Maximization

Maged F. Barsoum
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109
mbarsoum@jpl.nasa.gov

Christopher Jones
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109
crjones@jpl.nasa.gov

***Abstract*— Traditional constellations are uniformally spaced. By giving up uniform spacing, constellations can be designed to have larger joint (i.e. overall) capacity or parallel decoding capacity. In this paper non-uniformally spaced (i.e. 'geometrically' shaped) constellations are designed to maximize either of these quantities. By way of numerical capacity computations we show that except in special cases, there are no universally optimal geometrically shaped constellations across all code rates, and that the optimization of a constellation has to target a specific code rate. Unlike joint capacity, optimizing for parallel decoding capacity is label dependent. For PAM and PSK constellations, we found the maximum parallel decoding capacity to be achieved using gray labels. However, for PAM constellations, not *all* gray labels can yield the highest parallel decoding capacity. The conventional wisdom of using a $(log_2(M)-1)/log_2(M)$ code rate with an M points constellation for bandwidth efficient communications, could be re-evaluated in light of the newly developed constellations. An optimized constellation is used with a state-of-the-art LDPC code and simulation results are presented. This paper also draws a distinction between the practically complex probabilistic shaping and geometric shaping and in fact proves under broad conditions, that any gain in capacity which can be found via probabilistic shaping can also be achieved or exceeded solely through geometric shaping.**

I. Introduction

Constellation design has not traditionally been parameterized by the target code rate. This paper, however, argues by way of mutual information computations, that there is nearly always link margin to be gained through tailoring the design of a constellation for a specific code rate.

Although the minimum distance between constellation points is indicative of the capacity of a constellation at relatively high SNR's only, historically, constellations have most often been designed to maximize such minimum distance [6]. Because it had been noted, that increasing the dimensionality of a constellation allowed for a larger minimum distance for the same constellation energy per dimension, several researchers addressed the problem of designing multi-dimensional constellations with good minimum distance properties e.g. [9][10][11][12][7][8]. Multi-dimensional constellations were often useful in the context of set-partitioning [5], for trellis-coded modulation systems. However, in all these efforts the fundamental capacity of the constellation remained limited by a design that targeted the minimum distance, a measure that is more indicative of the performance of an ucoded system, but not well indicative of the capacity of a constellation at the SNR's of actual operation particularly with today's modern codes.

Shaping techniques provide an alternative approach for improving energy efficiency. The term shaping is traditionally applied to techniques that attempt to approximate the capacity achieving Gaussian input distribution by transmitting some constellation points with greater frequency than others, an operation which poses several practical difficulties. An example of a more practical shaping mechanism is given by Raphaeli in [1]. In section V, we show under broad conditions that any gain in capacity that can be achieved by probabilistic shaping can also be achieved by equiprobable but non-uniformally spaced signaling i.e. geometrical shaping.

Sun et al. [13] showed that with asymptotically large M, equiprobable, but non-uniformly spaced signaling can achieve the Gaussian input distribution. This approach can be approximated for finite M in an attempt to mimic Gaussian disitrbutions as, for example, in [3][4]. However, approaches that mimic the Gaussian input distribution do not parameterize designs based on the code rate, and are also limited to the design of PAM constellations only. In this paper, we show that one can do much better if the target code rate is parameterized in the optimization, and the location of constellation points allowed to be optimized further. For example, Sommer and Fettweis [3] achieve a 0.8 dB gain in parallel decoding capacity for PAM-32 at rate-1/2, while an optimization targeting rate-1/2 for PAM-32 can achieve a 1.5 dB gain over standard PAM-32, as will be shown here.

II. Design Methodology

A multi-dimensional constellation can be parameterized by the number of dimensions and the number of constellation points. As will be seen, the optimal design of a constellation depends on the target code rate to be used with it. The number of user bits per dimension, η, is related to code rate, R, and the total number of real signaling dimensions, N_{dim}, as follows:

$$\eta = \frac{R(\log_2(M))}{N_{\text{dim}}}$$

Modern systems employing LDPC or turbo codes can operate near channel capacity. In systems where there are no belief propagation iterations between the decoder and the constellation demapper, the constellation demapper can be thought of as part of the channel. Since modern codes operate

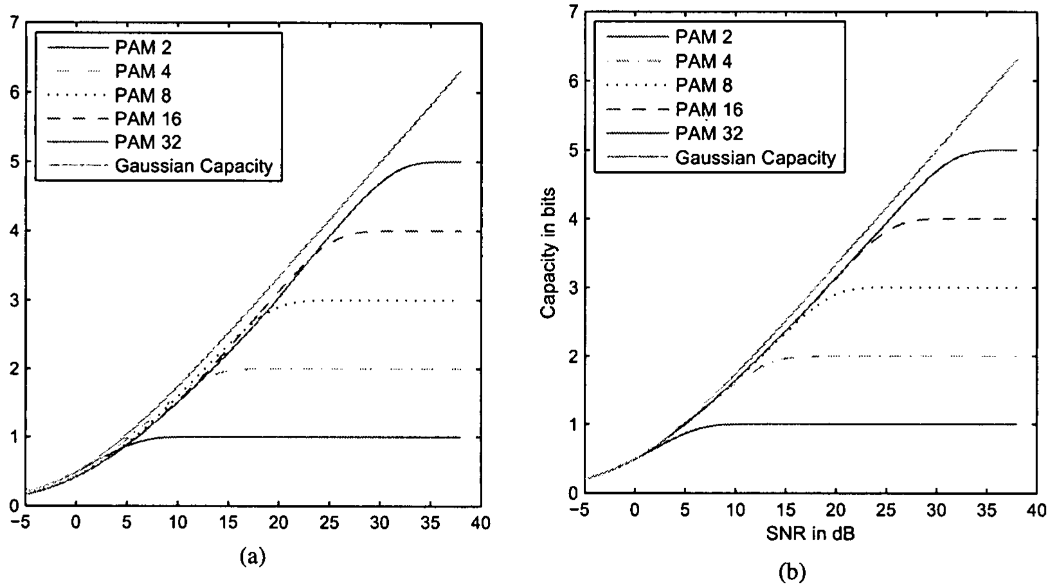

Fig. 1. (a) Parallel Decoding Capacity of PAM constellations together with the Gaussian Capacity curve. (b) Capacity of PAM constellations together with the Gaussian Capacity curve.

very close to the channel capacity, it would make sense to design constellations that maximize the channel capacity, again viewing the channel to include the demapper. For decoders of modern codes operating on bit likelihood ratios, the capacity of the channel defined as such is the parallel decoding capacity, given by

$$C_{PD} = \sum_{i=0}^{l-1} I(X_i, Y)$$

where X_i is the i^{th} bit of the l-bits transmitted symbol, and Y is the received symbol. With belief propagation iterations between the demapper and the decoder of a modern code, the demapper can no longer be viewed as part of the channel, and the joint capacity of the constellation becomes the tightest known bound on the system performance. However, practical evidence shows that the performance gain due to iterating between the demapper and the decoder is not significant when the code itself is a modern iterative code such as an LDPC or a turbo code. This phenomena can be seen in the simulation results of Section IV. For iterative codes, the parallel decoding capacity therefore remains a very good indicator of the system performance even with iterations between the demapper and the decoder.

While some researchers have previously attempted to optimize the bit labels of traditional constellations for one function or the other, the approach will take here is to jointly optimize the location of the constellation points and the labels. We do this by finding the optimal constellation points corresponding to the labels.

Besides designing constellations that maximize the parallel decoding capacity, we also design other constellations to maximize the joint capacity. Besides the theoretical value of joint capacity maximizing constellations, in some systems the capacity of the channel as viewed by the decoder is the joint capacity of the constellation. One example of such systems, is systems based on non-binary codes such as non-binary LDPC codes, that operate directly on non-binary symbols rather than bits.

In our optimization process, constellations were constrained using lower and upper bounds, with one constellation point fixed at the upper bound. This limits the search space without excluding the optimal solution. No additional constraints were placed on the energy of the constellation because the capacity computation kernel used signal-to-noise ratio, and adjusted the energy of the noise based on the energy of the constellation (the signal-to-noise ratio is defined in this paper as ratio of the constellation energy per dimension to the noise energy per dimension). It can easily be shown that an optimal constellation will have the property that the mean of all the constellation points is exactly zero. As such, we found that adding a zero mean constraint helps the optimization routine converge faster. To optimize at a particular user data rate, the process is iterative because the optimal SNR is not known at the beginning. The constellation is optimized for an initial SNR guess. The SNR at which this constellation gives the required user date rate is then used for finding a new constellation. The process repeats and the SNR's found every time converge. While there is no proof that the solutions found are globally optimum, repeating the optimization process with different starting random points increases the confidence in the optimality of the solutions found.

III. EXAMPLES

A. PAM Constellations

In this section, we will present different PAM constellations optimized for several user data rates. We start by presenting the joint capacity and the parallel decoding capacity for classic constellations at different SNR's. Figure 1 (b) shows the joint capacity for the classic PAM 2, 4, 8, 16 and 32, together with the Gaussian capacity. Figure 1 (a) shows the parallel decoding capacity for the same constellations together with the Gaussian capacity. To better view the differences between these curves at points close to the Gaussian capacity, we will instead plot the SNR gap to Gaussian capacity for different values of capacity for each constellation as shown in Fig. 2. It is interesting to note that unlike the joint capacity, at the same SNR, the parallel decoding capacity does not necessarily increase with the number of constellation points for classic constellations. Hereinafter, we will present the results of our optimized constellations using plots of SNR gap to capacity as done in figure 2.

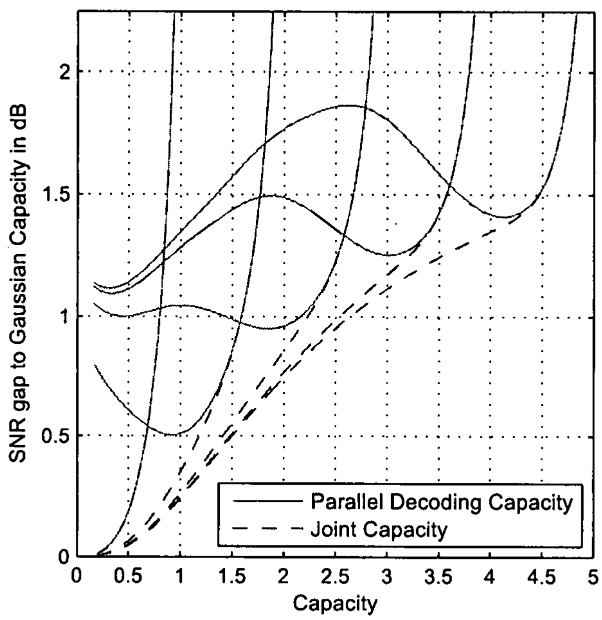

Fig. 2. SNR gap to Gaussian Capacity for the Joint Capacity and Parallel Decoding Capacity of 2,4,8,16 and 32 PAM constellations

1) PAM 2: For PAM 2, it is clear that the joint capacity will always be equal to the parallel decoding capacity, since there is only one bit to be transmitted. At any code rate, it is straightforward to show analytically that any optimal constellation will always have a mean of zero. For PAM 2, satisfying this condition implies that if one point is placed at x, the other one has to be placed at $-x$. The value of x can be viewed as a scaling factor that only changes the power of the constellation, but clearly won't changed the performance of the constellation at any fixed SNR. As such, the classic PAM 2 (or rather BPSK), is optimal at all code rates.

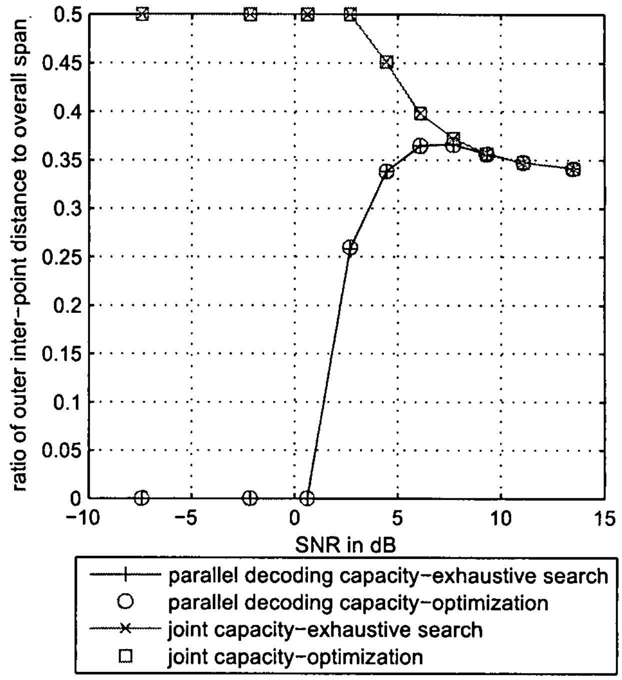

Fig. 3. r versus SNR for PAM 4 constellations optimized for joint capacity/parallel decoding capacity

2) PAM 4: We have designed 4 PAM constellations using two approaches. The first is an exhaustive search with a resolution of 0.002 with -1 and 1 as the lower and upper bounds on all the constellation points. The second is the optimization technique described earlier. The results agree.

At all user data rates, we found the optimized PAM 4 constellations to be symmetric around the origin. We will denote the location of the 4 points on the real line as $-a$, $-b$, b, a, from left to right. The constellation can therefore be fully described by the ratio

$$r = \frac{a-b}{2*a}$$

Figure 3 shows r for constellations optimized for joint capacity/ parallel decoding capacity at different SNR's. As seen in the figure, at low SNR's PAM 4 constellations optimized for joint capacity are very different from those optimized for parallel decoding capacity. A constellation optimized for joint capacity will have $b = 0$ i.e. it will look like a 3 points equally spaced constellation with two of the four symbols mapped to the same constellation point located at 0. A constellation optimized for parallel decoding capacity will have $b = a$ i.e. it will look like a BPSK constellation with two of the four symbols mapped to a and the other two mapped to $-a$.

As seen also in Fig. 3, at high SNR's it makes negligible difference whether a PAM 4 constellation is optimized for joint capacity or parallel decoding capacity. As the SNR increases, r approaches 1/3, i.e. the constellation becomes equally spaced. This illustrates that constellations designed to maximize the minimum distance between constellation points are well suited for high SNR's (as may be used, for example, in uncoded systems).

3) Other PAM constellations: After examining the design on PAM 4 constellations in detail in the previous section, we will now present a summary of the results for the design of PAM 8, 16, and 32. We have optimized the constellations for joint capacity and parallel decoding capacity for different target user bits per dimension (i.e. code rates). As already seen for PAM 4, optimized constellations are different depending on the target user bits per dimension and also depending on whether they have been designed to maximize the joint capacity or the parallel decoding capacity. For all the PAM constellations examined and at all code rates, we found that the labels that maximize the parallel decoding capacity are gray. However, not all gray labels can achieve the maximum possible decoding capacity even with the freedom to place the constellation points anywhere on the real line. Figure 4 (a) shows the SNR gap to Gaussian capacity for each constellation we optimized for joint capacity. Figure 4 (b) shows the SNR gap for each constellation we optimized for parallel decoding capacity. Again, it should be emphasized that each '+' on the plot represents a different constellation. Tables showing the optimized constellations could be given upon request for a patent application or any other purposes. While we have so far optimized PAM constellations and some PSK constellations only, the optimized constellations may include QAM, PSK, Multi-dimesional, as well as multi-dimensional spherical constellations.

B. PSK Constellations

In this section we will show sample results for PSK constellations. Figure 5 shows a 16 points PSK constellation optimized for joint capacity and another one optimized for parallel decoding capacity at SNR=8.87 dB. The constellation optimized for joint capacity is equally spaced while the one optimized for parallel decoding capacity isn't. However, it is gray labelled and has interesting symmetry properties. We conjecture that all PSK or even multi-dimensional spherical (i.e. equal magnitude) constellations optimized for joint capacity will always be uniformally spaced regardless of the code rate, they are optimized for.

IV. Performance with Modern Codes

In this section, sample simulation results of BICM systems using LDPC codes are presented. We compare the performance of two BICM systems, one using the classic 32 points PAM constellation and the other using our optimized 32 points PAM constellation. A rate $1/2$ AR4A LDPC code is used with a codeword size of 8 kbits and 16 kbits. Simulation results are presented in Fig. 6. As seen in the figure there is a significant SNR gap between the two systems (around 1.2 dB). It is also clear that performance does not improve much by iterating between the demapper and the decoder as pointed out earlier.

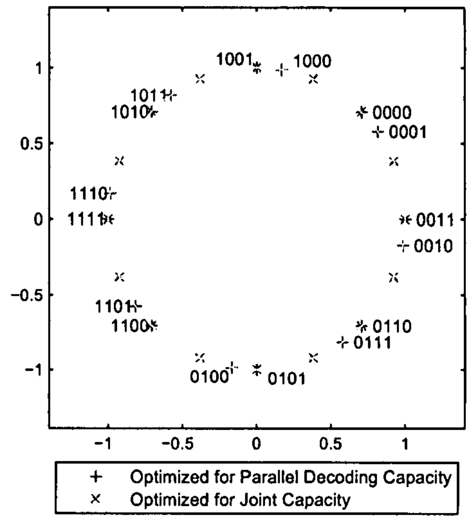

Fig. 5. A labelled 16 PSK constellation optimized for Parallel Decoding Capacity at SNR=8.87 dB along with a non-labelled constellation optimized for Joint Capacity at the same SNR

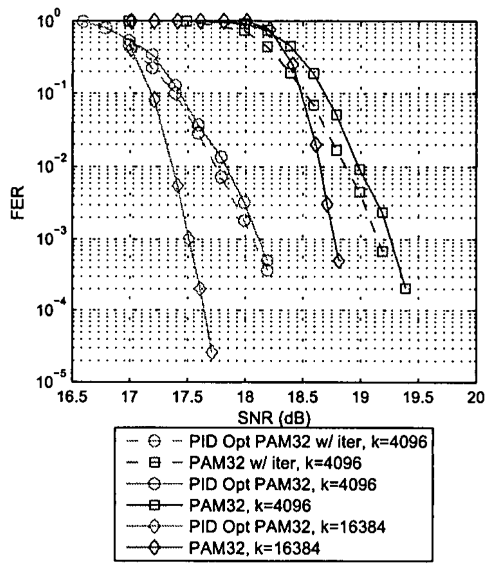

Fig. 6. Simulation results for a BICM system using a rate 1/2 LDPC code for the classic and the optimized PAM 32 constellation

V. Geometric shaping versus probabilistic shaping

Using the notation of [14] we compare the mutual information of systems with input alphabets $\mathcal{X}$ and $\mathcal{S}$ with elements x and s drawn from random variables X and S that exist in $\Re^n$ and have equal power constraints $EX^2 = ES^2 = P$. Symbols from one or the other alphabet are used to communicate across a channel that is specified only in so much that its output alphabet is given by $\mathcal{Y}$. Furthermore, the probability of symbols $x \in \mathcal{X}$ are constrained to be uniform $p(x) = \frac{1}{|\mathcal{X}|}$ while the probability of symbols $s \in \mathcal{S}$ form any valid

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase® Smarter legal research.