Function Modulation – the Theory for Green Modem

Subhendu Das, CCSI, California, USA, subhendu.das@ccsi-ca.com
Nirode Mohanty, Fellow-IEEE, CCSI, California, USA, nirode.mohanty@ccsi-ca.com
Avtar Singh, San Jose State University, California, USA, avtar.singh@sjsu.edu

Abstract - **Bandwidth of a digital communication channel is a valuable natural resource. Therefore it is important that the usage of this resource is minimized by carefully designing the digital modems. In this paper a new modulation scheme, called function modulation, is discussed that satisfies the Shannon's model and therefore can help to save bandwidth by creating a green modem. It is also shown that all existing modulation methods based on sinusoidal functions do not meet the requirements of Shannon's model and therefore have lower capacity for a given bandwidth.**

In addition, the paper shows that Shannon's capacity is also not the limit. This capacity was derived under the assumption that the symbol duration is infinity. By relaxing this infinite time requirement much higher capacity is achieved. In order to derive this new capacity result a powerful and very well known mathematical concept, the infinite dimensionality of function space is used. It is shown that this concept can be the foundation of the digital communication as well as digital signal processing engineering.

Keywords - information rates; sampling methods; orthogonal functions; communication; modulation.

I. INTRODUCTION

This paper is the extended version of [1]. Here all the relevant research work is integrated to present in details the function modulation (fm) concept and its capacity.

In the derivation of capacity theorem Shannon used general functions as digital communication symbols. It is shown in [1] that if sinusoidal functions are used then Shannon's conditions cannot be satisfied and therefore it will not be possible to achieve the capacity results derived in Shannon's paper [2]. The fm method uses this general class of functions and it is shown that this method can achieve the Shannon's capacity result.

Another very important assumption in the derivation of the original capacity theorem was that all symbols must be of infinite time durations. This infinite time for symbols is not feasible in engineering. By relaxing this condition it is shown that even higher capacity can be achieved, under the same channel bandwidth requirement, giving an opportunity to design a green modem.

Thus two major changes are introduced. First, the sinusoidal functions are replaced by the general class of functions to create the fm modems. This fm model is called the Shannon's model, because we show that his geometric proof is based on this general class of functions. Second, the infinite time assumption is replaced by the finite time assumption to derive a new high rate sampling theorem. The above two ideas are integrated in the fm system using a Software Radio (SWR) approach. This fm design is the green modem. This system gives a method for implementing Shannon's capacity theorem, which was missing from the communication literature.

A detailed discussion of a very well known mathematical theory of infinite dimensionality of function space is presented. Then this concept is used to derive the new sampling theorem. This theorem shows that [3] for finite duration signals more we sample more information we get from the signal, thus theoretically validating the common engineering practice. Like in the original paper [2], this new sampling theorem is used to derive the new capacity theorem. The new theorem shows that the capacity depends on the sample rate, and therefore can be much higher, theoretically unbounded, but practically bounded by the technology.

Shannon's original theory was derived using a geometric concept of n-dimensional Cartesian space where n approaches infinity. In this paper we use an apparently different geometric approach using infinite dimensional function space. We say it apparently, because on the face of it they are completely different, but their underlying concepts are linked together. We use our geometric approach to first derive the original result over finite time and then extend it to new result, also over finite time.

The paper is organized in several sections and their subsections. The major sections are Fundamentals, Sampling Theorem, Function Modulation, and Capacity Theorem. In the Fundamentals section we put the original thoughts that triggered this research. These thoughts were provided by various reviewers. The Sampling Theorem section answers – how many samples we need to describe a finite duration signal. The Function Modulation section describes the transmitter and the receiver. The Capacity Theorem section shows that the Shannon's theorem can be extended if we use finite duration symbols.

II. FUNDAMENTALS

In this section we collect all the fundamental ideas that will be used in the rest of the paper. They are related to infinite time assumption of classical theories, infinite dimensionality of function space, finite duration sampling needs, and the software radio concepts that are considered

crucial to the understanding of the thoughts that integrate all the research work presented in this paper.

A. Infinite Time Assumption

In this section we show that the assumption of infinite time duration for signals is not practical and is not necessary for our theories. In real life and in all our engineering systems we use signals of finite time durations only. Intuitively this finite duration concept may not be quite obvious though. Ordinarily we know that all our engineering systems run continuously for days, months, and years. Traffic light signaling systems, Global Positioning Systems (GPS) satellite transmitters, long distance airplane flights etc. are some common examples of systems of infinite time durations. Then why do we talk about finite duration signals? The confusions will be cleared when we think little bit and examine the internal design principles, the architecture of our technology, and the theory behind our algorithms. Originally we never thought that this question will be asked, but it was, and therefore we look here, at the implementations, for an explanation.

The computer based embedded engineering applications run under basically two kinds of Operating Systems (OS). One of these OS uses periodic approaches. In these systems the OS has only one interrupt that is produced at a fixed rate by a timer counter. Here the same application runs periodically, at the rate of this interrupt, and executes a fixed algorithm over and over again on input signals of fixed and finite time duration. As an example, in digital communication engineering, these signals are usually the symbols of same fixed duration representing the digital data and the algorithm is the bit recovery process. Every time a an analog symbol comes, the algorithm recovers the bits from the symbol and then goes back to process the next arriving symbol.

Many core devices of an airplane, carrying passengers, are called flight critical systems. Similarly there are life critical systems, like pacemaker implanted inside human body. It is a very strict requirement that all flight critical and life critical systems have only one interrupt. This requirement is mainly used to keep the software simple and very deterministic. They all, as explained before, repeat the same periodic process of finite duration, but run practically for infinite time.

The other kind of applications is based on the Real Time multi-tasking Operating Systems (RTOS). This OS is required for systems with more than one interrupts which normally appear at asynchronous and non-periodic rate. When you have more than one interrupts, you need to decide which one to process first. This leads to the concept of priority or assignment of some kind of importance to each interrupt and an algorithm to select them. The software that does this work is nothing but the RTOS. Thus RTOS is essentially an efficient interrupt handling algorithm. Thus RTOS is not unique and can be designed in your way.

These RTOS based embedded applications are designed as a finite state machine. We are not going to present a theory of RTOS here. So to avoid confusions we do not try to distinguish among threads, tasks, processes, memory management, and states etc. We refer to all of these concepts as tasks, that is, we ignore all the details below the level of tasks, in this paper. These tasks are executed according to the arrival of interrupts and the design of the application software. The total application algorithm is still fixed and finite but the work load is distributed among these finite numbers of tasks. The execution time of each task is finite also. These tasks process the incoming signals of finite time and produce the required output of finite size.

An example will illustrate it better. A digital communication receiver can be designed to have many tasks – signal processing task, bit recovery task, error correcting task etc. They can be interconnected by data buffers, operating system calls, and application functions. All these tasks together, implement a finite state machine, execute a finite duration algorithm, and process a finite size data buffer. These data buffers are originated from the samples of the finite duration signals representing the symbols. The transmitter of a digital communication system can also be implemented using similar principles.

We should point out that it is possible to design application systems which are combinations or variants of these two basic concepts. Most commercial RTOS provide many or all of these capabilities. Thus although all of the engineering systems run continuously for all time, all of them are run under the above two basic OS environment. Or in other words for all practical engineering designs the signal availability windows, the measurement windows, and the processing windows are all of finite time. For more details of real time embedded system design principles see many standard text books, for example [4, pp. 73-88].

The signals may exist theoretically or mathematically for infinite time but in this paper none of our theories, derivations, and assumptions will use that infinite time interval assumption. However, interestingly enough, to deal with the finite duration problem we have to use the well known mathematical concept of infinite dimensionality of function space. Thus somehow infinity appears to be inescapable. In the next subsection we explain this infinite dimensionality idea in details.

B. Infinite Dimensionality

We will use the following basic notations and definitions in our paper. Consider the class of all real valued measurable functions in $L_2[a,b]$, defined over the finite time interval [a,b]. We assume that the following Lebesgue integral (1) is bounded, i.e.

$$\int_a^b |f(t)|^2 \, dt < \infty \,, \quad \forall f \in L_2[a,b] \tag{1}$$

Then we define the L_2 norm as in (2):

$$\|f\| = \left[\int_a^b |f(t)|^2\, dt\right]^{1/2},\ \forall f \in L_2[a,b] \quad (2)$$

Then the following (3) definition can be used for metric d

$$d(f,g) = \|f-g\| = \left[\int_a^b |f(t)-g(t)|^2\, dt\right]^{1/2} \quad (3)$$

In addition (4) defines the inner product as

$$\langle f,g\rangle = \int_a^b f(t)g(t)dt \quad \forall f,g \in L_2[a,b] \quad (4)$$

Under the above conditions the function space, L_2 [a,b], is a Hilbert space. One very important property of the Hilbert space [5, pp. 31-32], related to the communication theory, is that it contains a countable set of orthonormal basis functions. Let $\{\varphi_n, n = 1,2,..\}$ be such a set of basis functions. Then the following (5) holds:

$$\langle \varphi_i, \varphi_j\rangle = \delta_{ij} = \begin{cases} 0 & if\ i \neq j \\ 1 & if\ i = j \end{cases} \quad (5)$$

And for any $f \in L_2[a,b]$ the Fourier series (6) can be written using (5) as:

$$f(t) = \sum_{i=1}^{\infty} a_i \varphi_i(t),\ \ \forall t \in [a,b] \quad (6)$$

The expression (6) really means that for any given $\varepsilon > 0$ there exists an N such that

$$\|f(t) - \sum_{i=1}^{n} a_i \varphi_i(t)\| < \varepsilon,\ \forall n > N \quad (7)$$

Since the functions in (5) are orthogonal, the Fourier series coefficients in (6) can be obtained using the Lebesgue integral (8) as shown below:

$$a_i = \int_a^b f(t)\varphi_i(t)dt\,,\ i = 1,2,... \quad (8)$$

We should point out to avoid any confusion that the Fourier series (6) in this paper is assumed to be defined over finite time duration. Although, a can be -∞ and b can be +∞. Also, the functions $\varphi_n(t)$ are general functions and not necessarily sinusoidal harmonic functions. The expression (6) may be called as the generalized Fourier series.

In this paper we will consider only continuous functions and their Riemann integrability. We note that the continuous functions are measurable functions and the Riemann integrable functions are also Lebesgue integrable. Thus the Hilbert space theory (1-8) and the associated concepts will still remain applicable to our problems. Actually, the Lebesgue integrable functions form an equivalent class, in the sense that there exists a continuous function whose Lebegaue integral is same as the Lebesgue integral of any one of the measurable functions in that equivalent class. By the way, the Riemann integral is the one we study in our high school calculus course.

We observe from (6) that to represent a function accurately over any interval we need two sets of data: (A) An infinite set of basis functions, not necessarily orthogonal and (B) An infinite set of coefficients in the infinite series expression for the function, similar to (6). That is, these two sets completely define the information content in a mathematical function. Thus the information is not a superficial concept; it has a very meaningful, practical, and mathematical definition as mentioned in this paragraph.

Equality (6) happens only for infinite number of terms. Otherwise, the Fourier representation in (7) is only approximate for any finite number of terms. In this paper ε in (7) will be called as the measure of approximation or the accuracy estimate in representing a continuous function. The Hilbert space theory (1-8) ensures the existence of N in (7) for a given ε. The existence of such a countably infinite number of orthonormal basis functions (5) proves that the function space is an infinite dimensional vector space. This dimensionality does not depend on the length of the interval [a,b]. Even for a very small interval, like symbol time, or an infinite interval, a function is always an infinite dimensional vector. However, the context in which this vector is defined is also very important, which is, the entire function space in this case.

It is not necessary to have orthonormal basis functions for demonstrating that the function space is infinite dimensional. The collection of all polynomial functions $\{t^n, n = 1,2,..\}$ is linearly independent over the interval [a,b] and their number is also countable infinity. These polynomials can be used to represent any analytic function, i.e. a function that has all derivatives. Using Taylor's series (9) we can express such a f(t) at t as:

$$f(t) = \sum_{n=0}^{\infty} \frac{f^{(n)}(c)}{n!}\,(t-c)^n \quad (9)$$

around the neighborhood of any point c. Thus the above polynomial set is also a basis set for the function space. Therefore using the infinite Taylor series expression (9), we prove again that a function is an infinite dimensional vector over a finite interval. Here the information is defined by the derivative coefficients and the polynomial functions.

It can be shown that a band limited function is also infinite dimensional and therefore carries infinite amount of information. Consider a band limited function f(t), with bandwidth [-W,+W]. Then f(t) is given by the following (10) inverse Fourier Transform (FT) [2]:

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} F(w)e^{iwt}dw \quad (10)$$

$$= \frac{1}{2\pi}\int_{-2\pi W}^{+2\pi W} F(w)e^{iwt}dw \quad (11)$$

In (11) t is defined for all time in (–∞, +∞), but the frequency w is defined only over [–W,+W], and it can take any value: integer, rational, or irrational frequencies, within that frequency range.

The expression (11) shows that the band limited function f(t) has uncountably infinite number of frequencies. Therefore f(t) is an infinite dimensional vector. This is true even when we consider a small interval of time for the function f(t). In that small interval the function still has all the infinite frequency components corresponding to the points in [–W,+W]. This is another way of showing that a band limited function is an infinite dimensional vector over a finite measurement window.

It should be pointed out here that a constant function f(t) = C, as an element of function space, is also an infinite dimensional vector. The only difference is that all sample values are same. In terms of Taylor series the coefficients for a constant function are {C,0,0,….}, which is an infinite dimensional vector. In the next subsection we discuss an all software approach for the design of digital communication systems. In the next subsection we prove infinite dimensionality in another way.

C. Finite Duration Sampling

The following is a very common way of expressing functions in mathematics. Let f(t) be a continuous function defined over L_2[a,b]. Assume that we divide the finite time interval [a,b] into n > 1 equal parts using equally spaced points $\{t_1, t_2, \dots, t_n, t_{n+1}\}$ where t_1=a and t_{n+1}=b. Use the following (12) notations to represent the t-subintervals

$$\Delta t_i = \begin{cases} [t_i, t_{i+1}), & i=1,2,..,n-1 \\ [t_n, t_{n+1}], & i=n \end{cases} \quad (12)$$

Define the characteristic functions (13) as:

$$X_i(t) = \begin{cases} 1, & t \in \Delta t_i \\ 0, & t \notin \Delta t_i \end{cases} \quad i = 1,2,..,n \quad (13)$$

In this case, the characteristic functions, $X_i(t)$ are orthogonal over the interval [a,b] with respect to the inner product on L_2 [a,b]. Because (14) given below holds.

$$X_i(t)X_j(t) = 0, \ i \neq j, \ \forall t \in [a,b] \quad (14)$$

Also define the simple functions (15) as

$$f_n(t) = \sum_{i=1}^{n} f(t_i)X_i(t) \quad \forall t \in [a,b] \quad (15)$$

Here $f(t_i)$ is the sampled value of the function f(t) at $t = t_i$. It is easy to visualize that $f_n(t)$ is a sequence of discrete step functions over n. Expression (15) is an approximate Fourier series representation of f(t) over [a,b]. This representation uses the samples of the function f(t) at equal intervals, $f_n(t)$ uses n number of samples. We show that this approximate representation (15) improves and approaches f(t) as we increase the number of samples, the value of n towards infinity.

Theorem 1: $f_n(t) \to f(t) \ as \ n \to \infty, \ \forall t \in [a,b]$

To prove the theorem, define (16) as the error expression

$$\Delta y_n = \max_t |f(t) - f_n(t)|, \ \forall t \in [a,b] \quad (16)$$

It is clear that $\{\Delta y_n\}$ is a monotonically decreasing sequence of n. Therefore, given any $\varepsilon > 0$ we can find an N such that $\Delta y_n \leq \varepsilon / \sqrt{(b-a)}$ for all n > N.

Starting with (17) derive the difference in norm:

$$\|f - f_n\| = \left[\int_a^b |f(t) - f_n(t)|^2 dt\right]^{1/2} \quad (17)$$

$$= \left[\int_a^b |f(t) - \sum_{i=1}^{n} f(t_i)X_i(t)|^2 \, dt\right]^{1/2} \quad (18)$$

$$= \left[\int_a^b |\sum_{i=1}^{n} f(t)X_i(t) - \sum_{i=1}^{n} f(t_i)X_i(t)|^2 \, dt\right]^{1/2} \quad (19)$$

$$= \left[\int_a^b \left[\sum_{i=1}^{n} \big(f(t) - f(t_i)\big)X_i(t)\right]^2 dt\right]^{1/2} \quad (20)$$

Now performing the squaring operation, noting that equation (14) holds, expression (21) helps to further simplify the expression (20):

$$= \left[\int_a^b [\sum_{i=1}^{n} [f(t) - f(t_i)]^2 X_i^2(t)] dt\right]^{1/2} \quad (21)$$

$$\leq \left[\int_a^b [\sum_{i=1}^{n} [\Delta y_n]^2 X_i^2(t)] dt\right]^{1/2} \quad (22)$$

$$= \left[\Delta y_n^2 \int_a^b (\sum_{i=1}^{n} X_i^2(t)) dt\right]^{1/2} \quad (23)$$

$$= [\Delta y_n^2 (b-a)]^{1/2} = \sqrt{(b-a)} \Delta y_n \leq \varepsilon \quad (24)$$

Thus from (17) we see that (25) holds, $\forall n \geq N$

$$\|f(t) - \sum_{i=1}^{n} f(t_i)X_i(t)\| \leq \epsilon, \forall t \in [a,b] \quad (25)$$

Which essentially means (26):

$$f(t) = \sum_{i=1}^{\infty} f(t_i)X_i(t), \ \forall t \in [a,b] \quad (26)$$

This concludes the proof of Theorem 1.

Theorem 1 proves that infinite sample rate is necessary to represent a continuous function correctly over a finite time interval. Theorem 1 is similar to the one described for measurable functions in [6, pp. 185-187]. However the coefficients are not sampled values in that theorem. Another proof can be found in [7, pp. 247-257] where the Bernstein polynomial has been used instead of the characteristic function.

The above theorem confirms a very well known engineering practice. In all engineering applications, our engineers always sample a signal at more than two to four times the Nyquist rate. Theorem 1 only mathematically justifies that well known practice. We will show that this theorem also provides the analytical foundation for our new capacity theorem. Thus Theorem 1, although very simple and obvious, has a profound implication in both digital

signal processing and communication theories. Note that Theorem 1 does not depend on the bandwidth of the function f(t). However ε and N, as used in (25), are dependent on the bandwidth.

Again from Theorem 1 we conclude that the function space is infinite dimensional and the information content can be represented by its infinite number of samples. This result is very important because these samples are generated by the Analog to Digital Converter (ADC) in our technology today. Thus the ADC actually produces the information content of a function.

D. Software RadioApproach

The foundation of the existing digital communication schemes is based on modulating the three parameters, amplitude A, frequency f, and phase φ of the sinusoidal function (27):

$$s(t) = A\sin(2\pi ft + \varphi) \quad (27)$$

Most of the existing methods vary, in discrete steps, one or more of the above three parameters to represent the digital data. These methods are known as Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), and Phase Shift Keying (PSK). Collectively we call them as Shift Keying (SHK) methods in this paper.

In this paper we present a modulation method for digital communication that does not use sinusoidal functions, does not use any one of the keying methods mentioned, and does not use any kind of discrete variations during the symbol intervals as well as at the inter-symbol interfaces. We show that the entire symbol stream, after concatenation, remains analytic i.e. smooth and continuous.

In the proposed method we modulate the complete function *s(t)* or the *sin* function itself. Since we are modulating the functional structure of the expression *s(t)*, we call it a function modulation (fm) method. Also, since our method does not use any discrete changes in the waveform or the function representing the symbol we call it an analog approach.

Our design approach is based on the concept of Software Radio (SWR), where we use batch data-in and batch data-out processing method as opposed to more conventional sample-in and sample-out type real time method. Of course we do this only for one symbol time which is usually millisecond or microsecond long. This SWR approach allows us to see the past, the present, and therefore the entire history of the data simultaneously, and to help extract information more effectively at the receiver. Observe that the symbol duration does not always indicate capacity or bit rate. Same symbol duration can be used for transmitting 10 bits or 20 bits of data as explained in our function modulation section. It is the number of symbols that dictates data rate or capacity. From our theory you will be able to find out that nanosecond is quite meaningful also and with our present day technology.

In this SWR method, since we are not using sinusoidal functions, we do not need to use Voltage Controlled Oscillators (VCO), Phase Lock Loops (PLL), up down converters etc., which are dependent on the concept of sinusoidal functions. We also do not need to use any linear feedback control system type concepts in this batch data processing method. Thus we avoid all the instability problems related to linear feedback theory.

This SWR design can be implemented entirely on standard off-the-shelf Digital Signal Processors (DSP) using programming concepts of time domain approaches. The advent of modern high speed DSPs enables us to take this approach. We will see that this time domain approach is more reliable and have meaningful theoretical foundation as opposed to Laplace transform or Fourier Transform (FT) based concepts that require linearity and infinite time assumptions. It should be understood that custom Application Specific Integrated Circuits (ASIC) instead of an off-the-shelf DSP can be more powerful for mass production.

Thus in our software radio approach we do not implement electronic hardware concepts using software languages. This approach allows us to implement our thought process using software languages. The thought process should not depend on the technology. The modern DSPs, high speed Analog to Digital Converters (ADC), and Digital to Analog Converters (DAC) allow us to implement our thoughts the way we think. An example of our thought process can be like this: "We want to send hundreds of symbols and our receiver should be able to detect them." We show that we have just implemented the above thought using our DSP concept.

Thus this SWR approach is not just moving the sampler near the antenna, and then implementing hardware logic using programming languages; it is all about rethinking the entire concepts using DSP, it is a paradigm shift. We do not treat things using moment by moment, like in sample by sample approaches. We treat the entire life history together, using batch data process, along with global system level, and simultaneous concept, as described later to implement our SWR. This kind of time domain approach has many advantages over conventional transform domain approaches. We know that the transform methods make infinite time and linearity assumptions. Both assumptions are not realistic in engineering problems and therefore will provide suboptimal solutions. Our approach will allow us to use mathematical techniques that we have never used before in real time, like integral equations, least square etc.

There is another very important common feature of all the existing digital communication schemes - bits of the data stream do not play any role, in the following sense, in the design of communication systems. For example, in an m-bit data stream we consider $M=2^m$ data packets. The internal bit pattern of each packet is not important. Only the number M is important. Existing methods select M waveforms to represent each one of these M packets. Any one of the

waveforms can be used to represent any one of the M data packets. Thus the link is very artificial. We could even send names of M authors, or titles of M books using these M waveforms. In the proposed method, the bits are directly embedded in the symbol i.e. there is a direct physical link between the bit pattern and the wave shape of the symbol. Thus we are sending bits also and not just symbols. Therefore Bit Error Rate (BER) is directly meaningful.

We introduce an intermediate space, called bit function space, between the data space and the symbol space. The dimension of the bit function space is m, the number of bits in the data packets. We show that fm receiver does not need to search in the symbol space; instead we can search in this newly defined bit function space. This bit function space approach reduces the number of searches. We show that, for the orthogonal bit function case, we need to search over only m bit functions instead of 2^m symbols as required by Shannon's capacity model. This is a significant reduction in the complexity of the fm receiver design and has the potentiality of providing high capacity systems.

Thus our software radio approach is dramatically different from the existing digital communication engineering. We have taken a new look to communication engineering from a global perspective and integrated the power of hardware, software, algorithm, and system level technologies. Next we extend the sampling theorem.

III. SAMPLING THEOREM

We show that when the signals are defined over finite time interval then the Nyquist rate is not enough for signal reconstruction. We must sample as fast as we can or as much the technology permits.

A. Finite Duration Case

Consider the sinusoidal function (28):

$$s(t) = A \sin (2 \pi f t + \theta) \quad (28)$$

We can see from the above expression that a sinusoidal function can be completely specified by three parameters A, f, and θ. So we can use (29) to express a sine function as a three dimensional vector:

$$s = [A, f, \theta] \quad (29)$$

However (29) is very misleading. There is a major hidden assumption; that the parameters of (29) are related by the sine function. Therefore (30) can be used to give a more precise representation of (29):

$$s = [A, f, \theta, \text{“sine”}] \quad (30)$$

The word sine in (30) means the Taylor's series, which has an infinite number of coefficients. Therefore when we say (29) we really mean (30) and that the sine function (28), as usual, is really an infinite dimensional vector.

Now assume, without loss of generality, that (28) is defined over one period. That is, we have collected the signal from the display of a digital oscilloscope. We can then use the following three equations (31) to solve for the three unknown parameters, A, f, and θ:

$$\begin{aligned} s_1 &= A \sin (2 \pi f t_1 + \theta) \\ s_2 &= A \sin (2 \pi f t_2 + \theta) \\ s_3 &= A \sin (2 \pi f t_3 + \theta) \end{aligned} \quad (31)$$

where t_1 , t_2 , t_3 are sample times and s_1 , s_2 , s_3 are corresponding three sample values. Again (32) gives a more meaningful representation in terms of samples:

$$s = [(s_1,t_1), (s_2,t_2), (s_3,t_3), \text{“sine”}] \quad (32)$$

Hence with the sinusoidal assumption, a sine function can be completely specified by only three samples. The above analysis gives a simple proof of the original sampling theorem. For band limited functions we can consider this sinusoid as the highest frequency sine wave in the signal. We can now state the well known result:

Theorem 2: A sinusoidal function, with sinusoidal assumption, can be completely specified by three non-zero samples of the function taken at any three points in its period.

From (31) we see that if we assume sinusoidality then more than three samples, or higher than Nyquist rate, will give redundant information. However without sinsoidality assumptions more sample we take more information we get, as is done in common engineering practice. It should be pointed out that Shannon's sampling theorem assumes sinusoidality. Because it is derived using the concept of bandwidth, which is defined using Fourier series or transform, which in turn uses sinusoidal functions.

Theorem 2 says that the sampling theorem should be stated as $f_s > 2f_m$ instead of $f_s \geq 2f_m$ that is, the equality should be replaced by strict inequality. Here, f_m is the signal bandwidth, and f_s is the sampling frequency. There are some engineering books [8, p. 63] that mention strict inequality.

Shannon writes about his sampling theorem [2, p. 448] in the following way: “If a function f(t) contains no frequencies higher than W cps, it is completely determined by giving its ordinates at a series of points spaced 1/2 W seconds apart.” The proof [2] is very simple and runs along the following lines. See also [9, p. 271]. A band limited function f(t) can be written as in (26). Substituting t = n/(2W) in (26) we get the following expression (33):

$$f\left(\frac{n}{2W}\right) = \frac{1}{2\pi}\int_{-2\pi W}^{+2\pi W} F(w) e^{iw\frac{n}{2W}}\, dw \quad (33)$$

Then the paper [2] makes the following comments: “On the left are the values of f(t) at the sampling points. The integral on the right will be recognized as essentially the nth coefficient in a Fourier-series expansion of the function

F(w), taking the interval –W to +W as a fundamental period. This means that the values of the samples f(n/2W) determine the Fourier coefficients in the series expansion of F(W).” It then continues “Thus they determine F(w), since F(w) is zero for frequencies greater than W, and for lower frequencies F(w) is determined if its Fourier coefficients are determined.”

Thus the idea behind his proof is that from the samples of f(t) we reconstruct the unknown F(w) using (33). Then from this known F(w) we can find f(t) using (10) for all time t. One important feature of the above proof is that it requires that the function needs to exist for infinite time, because only then you get all infinite samples from (33). We show that his proof can be extended to define functions over any finite interval with any degree of accuracy by increasing the sample rate. The idea is similar, we construct F(w) from the samples of f(t).

We use the principles behind the numerical inversion of Laplace transform method as described in [10, p. 359]. Let F(w) be the unknown band limited Fourier transform, defined over [-W,+W]. Let the measurement window for the function f(t) be [0,T], where T is finite and not necessarily a large number. Divide the frequency interval 2W into K smaller equal sub-intervals of width Δw with equally spaced points $\{w_j\}$ and assume that $\{F(w_j)\}$ is constant but unknown over that i-th interval. Then we can express the integration in (33) approximately by (34):

$$f(t) \approx \frac{1}{2\pi}(\Delta w)\sum_{j=1}^{K} e^{itw_j} F(w_j) \quad (34)$$

The right hand side of (34) is a linear equation in $\{F(w_j)\}$, which is unknown. Now we can also divide the interval [0,T] into K equal parts with equally spaced points $\{t_j\}$ and let the corresponding known sample values be $\{f(t_j)\}$. Then if we repeat the expression (34) for each sample point t_j we get K simultaneous equations in the K unknown variables $\{F(w_j)\}$ as shown below by (35):

$$\begin{bmatrix} f(t_1) \\ f(t_2) \\ \vdots \\ f(t_K) \end{bmatrix} = \frac{\Delta w}{2\pi} \begin{bmatrix} e^{it_1w_1} & e^{it_1w_2} \dots e^{it_1w_K} \\ e^{it_2w_1} & e^{it_2w_2} \dots e^{it_2w_K} \\ & \vdots \\ e^{it_Kw_1} & e^{it_Kw_2} \dots e^{it_Kw_K} \end{bmatrix} \begin{bmatrix} F(w_1) \\ F(w_2) \\ \vdots \\ F(w_K) \end{bmatrix} \quad (35)$$

These equations are independent because exponential functions in (34) are independent. Therefore we can solve them for $\{F(w_j)\}$. Theorem 1 ensures that the sets $\{F(w_j)\}$ and $\{f(t_j)\}$ can be selected to achieve any level of accuracy requirements in (34) for either f(t) or F(w).

For convenience we assume that the number of terms K in (34) is equal to Tkf_s which is equal to 2kWT. Here f_s is the Nyquist sample rate and k > 1. We state the following new sampling theorem.

Theorem 3: Let f(t) be a band limited function with bandwidth restricted to [-W,+W] and available over the finite measurement window [0,T]. Then given any accuracy estimate ε >0, there exists a constant k>1 such that 2kWT equally spaced samples of f(t) over [0,T] will completely specify the Fourier transform F(w) of f(t) with the given accuracy ε. This F(w) can then be used to find f(t) for all time t.

In a sense Shannon’s sampling theorem gives a sufficient condition. That is, if we sample at twice the bandwidth rate and collect all the infinite number of samples then we can recover the function. We point out that this is not a necessary condition. That is, his theorem does not say that if T is finite then we cannot recover the function accurately by sampling it. We have confirmed this idea in the above proof of Theorem 3.

Shannon proves his sampling theorem [2] in another way. Any continuous function can be expressed using the Hilbert space based Fourier expression (6). Shannon has used the above expression for a band limited function f(t), defined over infinite time interval. He has shown that if we use (36)

$$\varphi_n(t) = \frac{sin\{\pi f_s[t-(n/f_s)]\}}{\pi f_s[t-(n/f_s)]} \quad (36)$$

Then (37) will give the coefficients of (6):

$$a_n = f(n/f_s) \quad (37)$$

Thus (38) can be used to express f(t) [11, p. 58]:

$$f(t) = \sum_{n=-\infty}^{\infty} f(n/f_s) \frac{sin\{\pi f_s[t-(n/f_s)]\}}{\pi f_s[t-(n/f_s)]} \quad (38)$$

Here $f_s \geq 2W$, where W is the finite bandwidth of the function f(t). The set $\{\varphi_n\}$ in (36) is orthogonal only over $(-\infty,+\infty)$.

We make the following observations about (38):

- The representation (38) is exact only when infinite time interval and infinite terms are considered.
- If we truncate to finite time interval then the functions φ_n in (36) will no longer be orthogonal, and therefore will not form a basis set, and consequently will not be able to represent the function f(t) correctly.
- If in addition we consider only finite number of terms of the series in (38) then more errors will be created because we are not considering all the basis functions. We will only be considering a subspace of the entire function space.

We prove again that, by increasing the sample rate we can get any desired approximation of f(t), over any finite time

interval [0,T], using the same sinc functions of (36). From calculus we know that the following (39) limit holds:

$$\lim_{x\to\infty} \frac{\sin x}{x} = 0 \quad (39)$$

Assume that f_s is the Nyquist sampling frequency, i.e. f_s = 2W. Let us sample the signal at k times the Nyquist rate. Here k>1 is any real number. Then using (39), we can show that given any T and a small $\delta > 0$, there exists an N such that (40) as given below holds:

$$\left|\frac{sin(\pi k f_s t)}{\pi k f_s t}\right| < \delta, \forall k > N, \forall t \geq T \quad (40)$$

Thus these orthogonal functions (36) substantially go to zero outside any finite interval [0,T] for large enough sampling rate and still maintain their orthogonality property, substantially, over [0,T]. Thus by increasing the sample rate we squeeze many of these functions within this finite interval. The tails of these functions become substantially zero outside this interval as seen from (39). The squeezing situation is also shown in Fig. 1. Therefore for a given band limited function f(t), with signal capture time limited to the finite window [0,T], we can always find a high enough sample rate, kf_s so that given any ε>0 the expression (41) will be true:

$$\left\| f(t) - \sum_{n=0}^{K} f(\frac{n}{kf_s}) \frac{sin\{\pi k f_s[t-(n/kf_s)]\}}{\pi k f_s[t-(n/kf_s)]} \right\| < \varepsilon \quad (41)$$
$$\forall k > N, \forall t \in [0, T]$$

Observe that from the infinite duration Fourier series (38) we have derived a finite duration Fourier series (41) merely by increasing the sample rate. Our finite duration analysis is not just about recovering the signal, but finding how many samples are necessary to correctly represent the signal. Thus our focus is different from signal reconstruction theories, which is so well known in the literature.

The number of functions in the above series (41) is now K, which is equal to the number of samples over the period [0,T]. Thus K= kf_sT = 2kWT. As k increases the number of sinc functions increases and the distance between the consecutive sinc functions reduces thus giving higher sample rate. The original proof, [12, pp. 87-88] for (36-38), which is independent of sample rate, still remains valid as we increase the sample rate. That is, the sinc functions in (36) still remain orthogonal. It can be shown using the original method that the coefficients in (37) remain valid and represent the sample values. Thus the system still satisfies the Hilbert Space theory, making the expression (41) justified over [0,T]. Thus we can state the following new sampling theorem.

Theorem 4: Let f(t) be a band limited function with bandwidth restricted to [-W,+W] and available over the finite measurement window [0,T]. Then given any accuracy estimate ε there exists k>1 such that 2kWT equally spaced samples of f(t) over [0,T] along with their sinc functions, will completely specify the function f(t) for all t in [0,T] at the given accuracy.

Theorems 1 and 4 are identical, because the sinc function is the FT of the characteristic function. These theorems suggest that the Nyquist rate is not enough for finite duration signals. That is, we must sample as fast as we can depending on our technology and more we sample more information we get about the function. The expression (41) shows that the information content is in the samples and in the sinc functions. We mention again that our paper is not about signal reconstruction, it is about how many samples are required for a finite duration signal.

The paper [13] gives a good summary of the developments around sampling theorem during the first thirty years after the publication of [2]. Interestingly [13] talks briefly about finite duration time functions, but the sampling theorem is presented for the frequency samples, that is, over Fourier domain which is of infinite duration on the frequency axis.

In the following subsection we give a numerical example to show how higher rate samples actually improves the function reconstruction.

B. Numerical Example

We illustrate the effect of sample rate on the reconstruction of functions. Since every function can be considered as a Fourier series of sinusoidal harmonics, we take one sine wave and analyze it. This sine function may be considered as the highest frequency component of the original band limited signal. The Nyquist rate would be twice the bandwidth, that is, in this case twice the frequency of the sine wave. We are considering only one period, and therefore the Nyquist rate will give only two samples of the signal during the finite interval of its period. We are also assuming that we do not know or cannot use the analytical expression of the sine wave that we are trying to reconstruct.

Fig. 1 shows all the graphs of this numerical result. The figure will not be very readable on a printed paper. It is quite congested also. However it will be clear if enlarged on your computer. The horizontal axis represents the time in seconds. The full scale value is 0.001 seconds, that is, one millisecond. The vertical axis is normalized to unit value of amplitude. We thought it would not be a very good idea to break it down to seven new figures.

This figure has three groups 1a, 1b, and 1c represent the group for two samples case. Similarly 2a, 2b, and 2c represent another group for three samples reconstruction process. The last group consists of 3a, 3b, and 3c and shows the six samples results. In each group of Fig. 1 we show respectively, the sinc functions, reconstructed sine wave, the error between the actual sine wave and the reconstructed graph. The graphs show that the error decreases as we

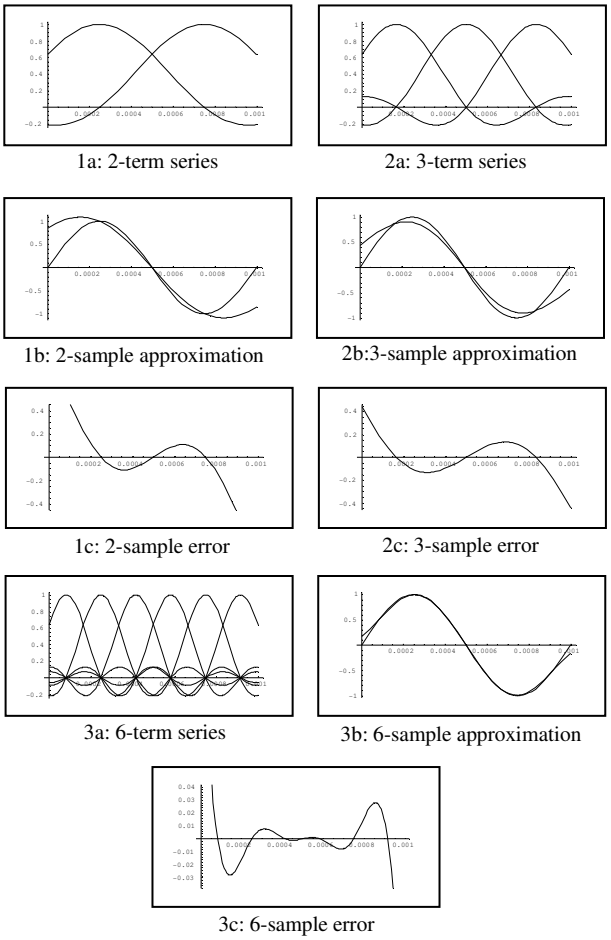

Figure 1. Signal reconstruction from samples

increase the sample rate. The middle set of graphs, 1b, 2b, and 3b, shows clearly how the signal reconstruction improves as we increase the number of samples. We have used formula (41) to reconstruct the sine waves from the sample values. A better method may give better result but our point is to show that more you sample better will be the recovery no matter what algorithm you use.

IV. FUNCTION MODULATION

A new modulation method called function modulation (fm) is discussed. We use the lower case letters fm to denote function modulation and reserve the upper case FM to represent Frequency Modulation method. We describe both transmitters and receivers of this fm scheme. The results of a practical implementation are discussed.

A. The fm Transmitter

Fig. 2 describes the design of a transmitter based on the function modulation (fm) method for digital communication system [14]. The left-hand-side vertical box represents a four bit data, as an element of data space, to be transmitted using one symbol *s(t)*. In Fig. 2 we have assumed, the number of bits, m, to be transmitted is four, as an example, without any loss of generality.

Let d = {d_i , i=1..m, d_i ε {0,1}}, be a column vector, and represent a data element in the data space. Here d_i represents the i-th bit of d with 0 or 1 as its value. Let G(t) = {g_i (t), i=1..m, t ε [0,T]}, also a column vector, represent a set of analytic and independent functions defined over the symbol interval [0,T]. We assign the i-th function $g_i(t)$ to the i-th bit location. In Fig. 2 the arrows from the bit locations to the bit function boxes define these one-to-one assignments. These functions are referred to as bit functions. The set G(t) defines the bit function space.

A set of functions G(t) is called dependent if there exists constants $\{c_i,\ i = 1..m\}$, not all zero, such that the expression given by (42) holds:

$$g_1(t)c_1 + g_2(t)c_2 + \cdots + g_m(t)c_m = 0 \qquad (42)$$

for all t ϵ [0,T]. If not then it is independent [15, pp. 177-181]. The above expression is a linear combination of functions. Here the coefficients $\{c_i,\ i = 1..m\}$ are all real numbers.

A real valued function is analytic if all derivatives are uniformly bounded [16, p. 238]. Analytic functions are band limited [12, p. 87]. An analytic function does not have any discrete jumps; it is a smooth and continuous function. In this paper the terms analog and analytic functions are used interchangeably.

The m bit functions are combined inside the algorithm box to produce one symbol function s(t). The collection of all symbols is called the symbol space. The Fig. 2 shows how we have introduced the concept of a bit function space in between the data space and the symbol space.

The algorithm is selected in such a way, so as to produce a symbol that is also an analytic function. For every bit pattern in the data space the algorithm produces one unique symbol in the symbol space using only m bit functions from the bit function space. The algorithm is an one-to-one and onto transformation, from the data space to the symbol space, ensuring that for every symbol it produces, there exists one unique bit pattern.

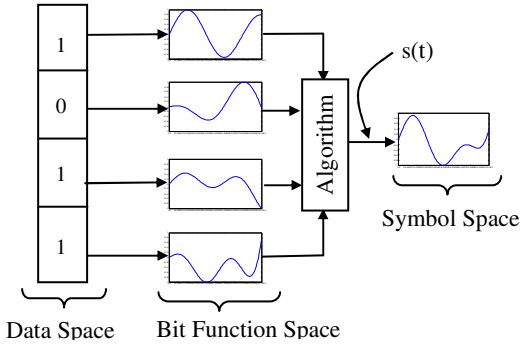

Figure 2. fm Transmitter

In general the algorithm in Fig. 2 can be represented by the following expression (43):

$$s(t) = A[G(t), d] \tag{43}$$

In (43) A is an arbitrary algorithm, operator, or transformation, which can be algebraic or dynamic, as well as, linear or non-linear. The operator A is a mapping from the product of the bit function space and the data space to the symbol space. A simple example of the operator A can be given by the expression (44):

$$\begin{aligned} s(t) &= G'(t)d \\ &= g_1(t)d_1 + g_2(t)d_2 + \cdots + g_m(t)d_m \end{aligned} \tag{44}$$

The notation ' indicates the transpose of the column vector G(t). Equation (44) is not a linear combination in the strict sense. The coefficients in (44) are not real numbers; they are only 0-1 integers. We will call this algorithm as a 0-1 addition algorithm. In this paper (44) will define the fm transmitter.

When all bits are zeros, expression (44) does not produce a meaningful symbol. To circumvent this problem we use a special, predefined analytic function, consistent with the technology presented here, to represent the symbol when all bits are zeros. The receiver will first test for the presence or absence of this special function, to detect the transmitted data corresponding to all zero bits, before using the standard algorithm discussed later.

We see that excepting the zero bits case the bit values are directly used to create the symbols in our fm method. In fm method the expression (44) or the general algorithm (43) physically imbeds the bit values in the symbol function. The bits directly modify the symbols as the processor creates the symbol one bit after another by adding the corresponding bit functions to the partially generated symbol. We do not arbitrarily assign the symbols to bit patterns generated by the bits in the data packets. However, these arbitrary assignments are also perfectly feasible and meaningful operation in fm case. All we are doing is that we are using analytic functions to bit patterns. We are choosing these analytic functions in such a way, as shown later, that the symbols remain analytic even at the inter-symbol interfaces.

On a side note, we observe that this 0-1 addition process has a very interesting mathematical consequence. This fm approach can be used to invert the algebraic addition process. That is, if we add two numbers, say 2 and 3 to produce 5, then given 5 we can find out which two numbers were added. Representing the integers 0 through 9 by 10 different continuous functions we can do this. Zero may be represented by zero function. This concept can generate interesting consequences in mathematical Group theory.

The fm transmitter concept may appear similar to the Orthogonal Frequency Division Multiplexing (OFDM) concept [17]. But there are some major differences. OFDM uses one or more of the three basic modulation methods (ASK, FSK, and PSK) from the existing technology, fm does not. OFDM method configures the spectrum bandwidth into disjoint regions but fm does not. In fm every bit function spans the entire bandwidth of the channel. OFDM uses only harmonically related sinusoidal orthogonal functions; fm does not need to use any orthogonal function. fm can use non-sinusoidal orthogonal functions as well as non-orthogonal functions, OFDM cannot. However, under certain restrictive conditions OFDM is considered as a special case of fm technique. If we select only sinusoidal and harmonically related orthogonal bit functions, use only amplitude modulation with zero or full signal variation, and use 0-1 addition algorithm, then fm is same as OFDM.

B. The fm Receiver

At the receiver we will receive the symbol function s(t) as generated in Fig. 2, corrupted by the noise and/or the nonlinearities of the communication channel. Our objective at the receiver will be to find out which bit functions from the bit function space G(t) were used to generate the received symbol. That is, we have to decompose the received symbol into the component bit functions. The presence or absence of a bit function in the received symbol will indicate 1 or 0 value, respectively, for the bit at the corresponding bit location.

A set of bit functions G(t) is orthogonal if the following integral (45) holds:

$$\int_0^T g_j(t)g_i(t)dt = 0,\ i \neq j,\ i,j = 1..m \tag{45}$$

Observe that we have defined orthogonality over finite time interval [0,T].

All sinusoidal functions are orthogonal over infinite time interval. Only harmonically related sinusoidal functions are orthogonal over a finite time interval. It is easy to verify, using the above relation, that the two sinusoidal functions with frequencies 1000 Hz and 1100 Hz are not orthogonal over the period 1/1000 seconds or 1/1100 seconds. It is also well known [18] that there are infinitely many, band-limited, non-sinusoidal, orthogonal functions over a finite time interval. However there are only a finitely many band limited sinusoidal orthogonal functions over a given finite time interval.

The 0-1 addition formula gives the expression for the received symbol, *r(t)*, as shown below. The following (46) is a derived from (44).

$$r(t) = g_1(t)x_1 + g_2(t)x_2 + \cdots + g_m(t)x_m + w(t) \tag{46}$$

Here, w(t) is an Additive White Gaussian Noise (AWGN) process, $\{x_i\}$ are the bit values, unknown to the receiver but known to the transmitter and are equal to $\{d_i\}$. Thus x_i can be 0 or 1 only. If we assume that the bit functions in G(t) are orthogonal then we can find x_i using the following simple relation (47):

$$x_i = \int_0^T r(t) g_i(t)\, dt + w_i \ , \quad i = 1..m \tag{47}$$

In above w_i is the projection of *w(t)* over g_i *(t).* We can set the bit values d_i using the relation (48) given below:

$$d_i = \begin{cases} 1, & x > 0 \\ 0, & otherwise \end{cases} \qquad i = 1..m \tag{48}$$

A fm receiver design [19] that uses orthogonal functions is shown in Fig. 3.

From the above receiver figure we can also see that we are really detecting the bits. Every output line gives the values of a bit of the entire bit pattern that we have transmitted using one symbol. If we fail to detect one of the functions in the decomposition process, then we will make errors in the detection of that bit. Thus real Bit Error Rate (BER) will happen in this fm method. It is not that we are converting the symbol detection error into BER using some artificial relations.

Fig. 3 is identical to a standard figure in many communication textbooks. However it has a few significant differences also. Notice that it has only m parallel paths as opposed to 2^m parallel paths found [20, p. 135] in the existing methods. The output from each correlator is the bit value of the corresponding bit location, which is not the case in conventional methods. In conventional methods only one of the boxes produces an output indicating a symbol match in the corresponding path. You can also find a similar figure in textbooks that use orthogonal functions [20, p. 135] and that has m parallel paths. In that figure the output of each path is a real number. In Fig. 3 the outputs are only 0 or 1 integers representing the actual bit values.

Fig. 3 and equation (46) show that for the fm method, based on orthogonal functions, we need to search over only m functions in the bit function space as opposed to 2^m symbols in the symbol space. Thus the orthogonal fm method can significantly reduce the complexity of the receiver design. The introduction of the bit function space in between the symbol space and the data space helps us to achieve this interesting result. This concept indicates that the fm method has very high capacity. The dimension of the bit function space is m, because this space has m independent functions. On the other hand the dimension of the symbol space is 2^m because it is defined by the 2^m number of independent symbols.

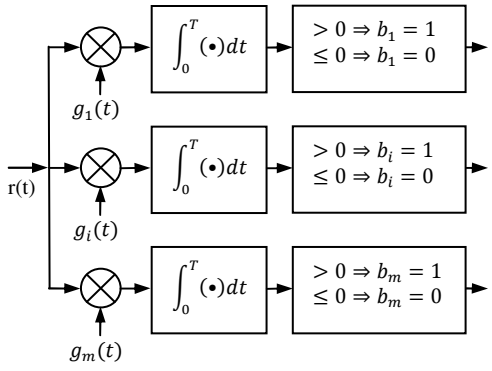

Figure 3. fm Receiver with orthogonal functions

The receiver design for the non-orthogonal case is quite complicated and involved. The design is not unique also. In the remaining part of this section we describe one numerical algorithm or approach for the design of a fm receiver that uses 0-1 addition algorithm and non-orthogonal set G(t). All the bit functions, $\{g_i(t)\}$, are in this case analytic and independent only. The set G(t) is known to both the receiver and the transmitter. Given the information in (46) our problem at the receiver is, again, to solve (46) for 0-1 integer values for the unknown variables $\{x_i\}$.

Note that, in this formulation, the problem (46) is not really a classical 0-1 Integer Programming Problem (IPP). There is no optimization function associated with the equality expression (46). There is a random noise variable in (46) which is also not found in the standard IPP. Also the coefficients in (46) are not real numbers but functions of time.

There are various methods available in the scientific and engineering literature for solving the above receiver problem. In this paper we discuss only one of them. We convert the problem (46) to a least square solution problem by sampling, at fixed intervals, all the signals n times over the symbol period [0,T], where n is an integer greater than or equal to m. Thus (46) can be expressed by the set of simultaneous equations (49):

$$\begin{bmatrix} r(t_1) \\ r(t_2) \\ \vdots \\ r(t_n) \end{bmatrix} = \begin{bmatrix} g_1(t_1) & g_2(t_1) & \dots & g_m(t_1) \\ g_1(t_2) & g_2(t_2) & \dots & g_m(t_2) \\ \vdots & \vdots & & \vdots \\ g_1(t_n) & g_2(t_n) & \dots & g_m(t_n) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} + \begin{bmatrix} w(t_1) \\ w(t_2) \\ \vdots \\ w(t_n) \end{bmatrix} \tag{49}$$

Here, $\{t_1, t_2, \dots t_n\}$ are equally spaced sample points inside the time interval [0,T]. We will assume n is larger than m giving us more equations than the number of unknown variables m. Using the matrix notation the problem defined by (49) can then be rewritten as in (50).

$$r = Ax + w \tag{50}$$

Since G(t) is a set of functions with analytical expressions they can be sampled any number of times. This is assured by the sampling Theorem 1. The length of the vector r can also be increased by interpolation between real samples obtained from Analog to Digital Converters (ADC). Thus the number of samples need not depend on the sample rate of the ADC or on other electronics in the receiver. This is one of the advantages of using the software radio approach discussed before. We can always get more number of equations than the number of unknowns giving us a better least square solution for (50).

In (50) r and w are n-column vectors with components consisting of n samples of the functions r(t) and the AWGN process w(t), respectively. A is a nxm rectangular matrix

with elements defined by (51), and x is the unknown 0-1 column vector [x_1, x_2,… x_m]' taken from {x_i }.

$$a_{ij} = g_j(t_i), i = 1..n, j = 1..m, t_i \in [0,T], n > m \quad (51)$$

Since the functions in the set {g_i (t)} are independent the matrix A with elements defined by (51) is a full rank matrix. Therefore A'A is non-singular and the real valued solution of (50) can be expressed using the pseudo inverse P of A [21] as given by (52):

$$x = Pr = (A'A)^{-1}A'r, \ where \ P = (A'A)^{-1}A' \quad (52)$$

The bit values {d_i} can then be obtained by the decision logic (53):

$$d_i = \begin{cases} 1, & x_i > \beta \\ 0, & otherwise \end{cases} \qquad i = 1..m \quad (53)$$

The threshold value β is a given constant representing the channel characteristics.

The pseudo inverse gives a least square error solution of the simultaneous linear equation (50). It essentially curve fits the received symbol function r(t) using the bit functions of the set G(t). Note that the matrix P is constant for a given fm system, that is, when G(t) is given. Therefore it is known to the receiver and can be precomputed and stored in memory. The accuracy of the fm system can be controlled by controlling the number of samples, n, for each function. The number of samples does not have to be the real samples from the ADC device. We can create an analytic function to interpolate the ADC samples and then derive as many samples as we want from the analytic expression. More details of the fm receiver with non-orthogonal functions can be found in [13].

C. Fourier Series and the fm Concept

The fm system can be considered as an implementation of the Fourier series. This interpretation will reveal many features of fm system. As we have mentioned before any continuous function defined over finite time interval can be expressed [5] by the Fourier series (54):

$$f(t) = \sum_{i=1}^{\infty} a_i \varphi_i(t), \ \forall t \in [0,T] \quad (54)$$

Where {φ_i(t)} satisfying (55)

$$\langle \varphi_i, \varphi_j \rangle = \delta_{ij} = \begin{cases} 0 & if \ i \neq j \\ 1 & if \ i = j \end{cases} \quad (55)$$

are orthonormal functions and defined over [0,T]. From (54) a finite term Fourier series (56) can be expressed as:

$$f(t) = \sum_{i=1}^{m} a_i \varphi_i(t), \ \forall t \in [0,T] \quad (56)$$

We also know that given the orthonormal functions {φ_i(t)} and the function f(t) we can find the real valued coefficients {a_i} of (56) using the following equation (57)

$$a_i = \int_0^T f(t)\varphi_i(t)dt \ , \ i = 1..m \quad (57)$$

If we now assume that {φ_i(t)} in (56) are the given orthonormal bit functions of the fm method, and {a_i} are the 0-1 bit values of the data pattern, then using (56) we can generate the function f(t), which can be considered as a symbol for the bits {a_i}. Thus we see that the fm transmitter of Fig. 2 actually implements a finite term Fourier series given by (56). Also we can see from (56) that there is no limit on how many bits we can transmit using orthogonal fm transmitter. Again, this explanation shows that the fm method has very high capacity.

The limit in (56) is defined by the number of band limited orthonormal functions that can be found. The paper by Slepian [18] assures that there are infinite numbers of such functions. In the later sections we show that the capacity of a digital communication channel is indeed unbounded and limited only by our technology of the receiver.

In this context we also point out that it is not necessary to use orthonormal functions in the Fourier series expression (56). The series (56) can be valid even if we use independent functions, 0-1 coefficients, and finite number of terms. We will still be able to extract the bits as shown before. Thus we have proven that the very general class of functions can be used for digital communication. Now we show how we have implemented the fm system in real life using real hardware.

D. A Real Life Implementation

The fm system has been tested [13] in a real engineering environment. In this section we briefly describe the hardware board, experimental setup, and the results of this practical real life test. We also discuss our global or system level approach to signal processing.

The block diagram of this off-the-shelf hardware boards that we found is described by Fig. 4. These are two identical TMS320C5402 DSP boards [22] from Texas Instruments (TI). Each board has a telephone line interface with a Data Access Arrangements (DAA) Integrated Circuit (IC). This DAA takes care of the voltage conditions and protection of the telephone line. The TLC320AD50 is an IC codec and contains both an Analog to Digital Converter (ADC) and a Digital to Analog Converter (DAC). This IC is the interface between the DSP and the analog world.

The board has a printer parallel port for interfacing with the computer. Via this printer port we control the boards using the TI Code Composer Studio (CCS) [23] software development tools. These boards allow us to perform the real time experiment on the Plain Old Telephone System (POTS) network.

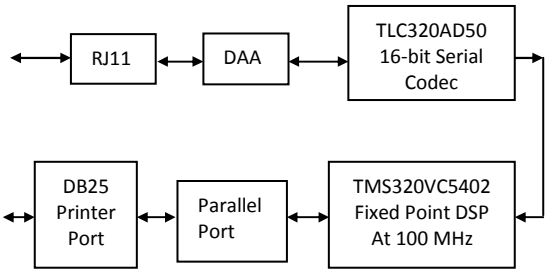

Figure 4. HW board functional block diagram

For this experiment we operate the codec for both receiver and transmitter at 16 kHz sample rate with 16-bit data resolutions. The symbol duration used was one millisecond. Only the transmission and reception of symbols via the telephone line are performed in real time using the hardware boards. The symbol creation and the symbol analysis functions are performed off line using Mathematica and Matlab software tools.

The experimental setup is shown in Fig. 5. We assemble everything, both transmitter and receiver, in one laboratory room with two telephone sockets having two different telephone numbers. The two DSP boards are connected to the telephone lines and to two computers to control them.

The bit functions shown in Fig. 6 and Fig. 7 are used to generate the symbol corresponding to the bit pattern 1011. This transmitted symbol is shown in Fig. 8. We transmit this symbol using our laboratory setup and capture the received signal, shown in Fig. 9, at the receiver end.

To synchronize the received signal we perform linear interpolation and up sampling. These two activities actually increase the resolution of the function. Synchronization is really very simple in batch data processing approach using a digital signal processor. This approach does allow you to do what you think and can visualize in your imaginary eyes. If you see the symbol on your oscilloscope, and think what you want do to it, you can do exactly that in real time on the computer memory buffer. Remember that there are no clocks and PLL involved here. Using this method we find

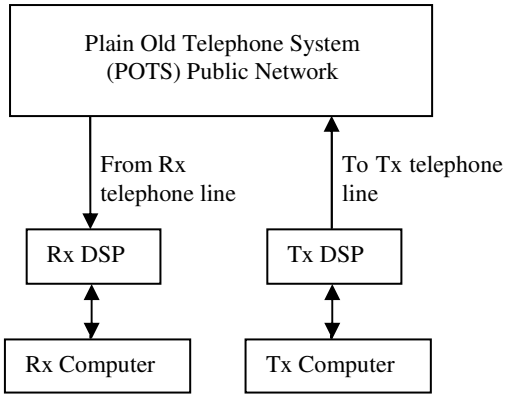

Figure 5. fm validation system

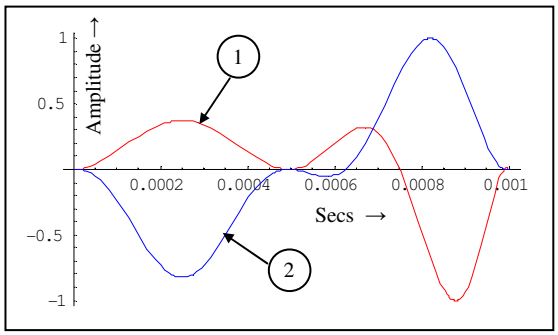

Figure 6. Bit-Functions 1 and 2

the exact zero crossing points by removing the required number of samples from both ends to make the received symbol start and end very close to the time axis. Finally we use (52) to solve for the least square curve fitting problem for 17 samples. The result is the following values for the unknown variables $\{x_i\}$:

$$\{1.8653, 0.45037, 1.03662, 0.851587\}$$

A threshold value of 0.5 for β in (53) gives the bit values for correct transmitted data, 1011.

Even though we encounter severe non-linear distortions we are still able to recover the bits correctly using only 17 samples. The experiment shows that the curve fitting method, using the bit functions along with the 0-1 addition algorithm, is indeed very robust. Note that it is also the availability of the entire data history that played a very important role in extracting the information.

As we can see from the figures the received signal has two positive peaks as opposed to three positive peaks in the transmitted signal. As if the second trough of the transmitted signal got folded up in the received signal.

It is clear that the conventional signal recovery methods, that use local concepts, no matter how many samples we take, cannot bring the received signal back to the transmitted form. However, a global approach or a systems approach, where we use the knowledge of the entire system can definitely help. We used the same sine wave frequencies of the transmitter, to interpolate the samples at the receiver. Here, of course, the high sample rate played an

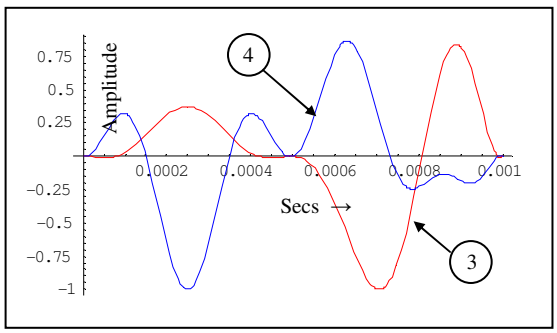

Figure 7. Bit-Functions 3 and 4

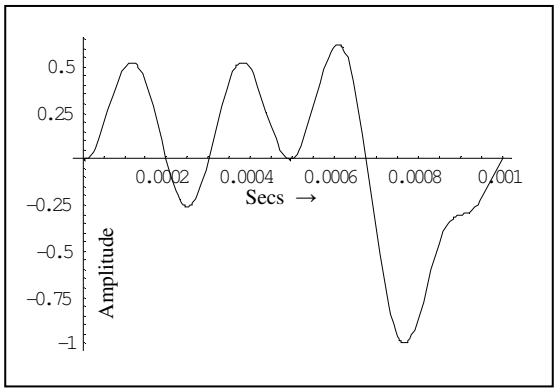

Figure 8. Transmitted fm symbol

important role in the least square interpolation method. The details of the signal processing, is quite involved, and is not given here. The large sample rate and the systems approach helped us to bring the received signal back to a shape that is very close to the transmitted signal, as shown in Fig. 10, which allowed us to detect the bits correctly. We can see that a total system level or global approach in signal processing can perform real miracles.

It should be mentioned that we performed these experiments over long period of time. During this trial and error period we developed the algorithm and the software discussed in this paper. In these lab experiments a series of concatenated symbols were transmitted using many different kinds of bit functions. In this paper we presented a specific case in a simple form for clear explanation. With a better hardware and software the algorithm presented here can be easily implemented online. Once the new hardware and the embedded environment become available we will present the results of real time high speed case. We are working on that direction now. Understandably, we are at a very early stage in terms of our real life implementation for a marketable product.

E. The fm Characteristics

The Power Spectral Density (PSD) of symbol s(t) in

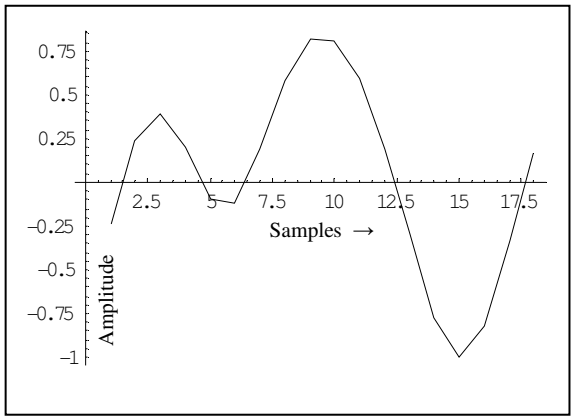

Figure 9. Received fm symbol

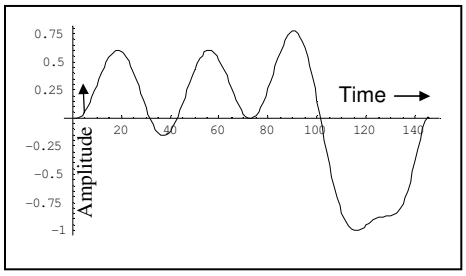

Figure 10. Best fit of the received symbol

Fig. 2 and the Bit Error Rate (BER) expressions have been derived. The BER has been derived for the case of orthogonal functions only. The PSD expression is valid for both orthogonal and non-orthogonal functions as bit functions.

The expression (58) generates the complete symbol stream *s(t)* for the 0-1 addition algorithm and for all time t of the infinte interval:

$$s(t) = \frac{1}{m}\sum_{k=-\infty}^{\infty}\sum_{i=1}^{m} d_i(k)g_i(t-kT) \tag{58}$$

The factor *m* was introduced to normalize the amplitude of the sysmbol. Since each bit function is normalized, the addition of *m* bit functions requires renormalization by *m*. Note that d_i is the bit value, 0 or 1, and is not the multilevel value of the data elelment even though we are considering *m*-bit data. Also note that there is no constant pulse shaping signal associated with the symbol stream. The bit functions replaced them. It is interesting to observe that the structure of the above mathematical expression is similar to that of the standard OFDM [17, pp. 5-8].

Define the bit correlation function by (59)

$$R_{kl}(p,q) = E[d_k(p)d_l^*(q)] \tag{59}$$

and its two sided FT by (60)

$$C_{kl}(v,w) = \sum_{p=-\infty}^{\infty}\sum_{q=-\infty}^{\infty} R_{kl}(p,q)e^{-j2\pi vpT}e^{-j2\pi wqT} \tag{60}$$

Using the above two definitions and substituting *G(f)* as the FT of *g(t),* the PSD can be represented by (61):

$$S(f) = \frac{1}{m^2}\sum_{k=1}^{m}\sum_{l=1}^{m} C_{kl}(-f,f)G_k^*(f)G_l(f) \tag{61}$$

Although (58) is similar to OFDM, the final expression (61) for S is quite different. Here we have reused the symbol G with different meaning and we hope that the context helps to prevent confusions, if any. The absence of periodic pulses in (58) removed all discrete terms from the expression (61).

As an example of the PSD result, we use the following sinusoidal functions as defined by (62), used in Multiple Phase Shift Keying (MPSK) systems, as the bit functions in fm system:

$$g_k(t) = A\cos(2\pi f_c t + \theta_k),\ -T/2 \le t \le T/2 \tag{62}$$

Substituting the above sine functions in (61) and performing some algebraic simplifications, we can arrive at the following PSD expression (63) for this case of fm scheme.

$$S_{fm}(f) = \left(\frac{AT}{2}\right)^2 \left(m \frac{\sin \pi f m T}{\pi f m T}\right)^2 \quad (63)$$

Comparing the above PSD expression with that of the MPSK [24] system, given below by (64),

$$S_{MPSK}(f) = A^2 T \left(\frac{\sin \pi f T}{\pi f T}\right)^2 \quad (64)$$

we see that the fm spectrum requirement is narrower as m increases. The PSD graphs of the two systems are plotted in Fig. 11. The graphs for fm system are plotted for four different bit data length m. If we use the band-limited orthonormal functions as designed in this paper, then the bandwidth requirements can be further reduced. It is also well known [25] that the band-limited functions do not create inter symbol interference.

We summarize the BER result here for the orthogonal case only. Consider one of the parallel paths of the orthogonal fm detection method presented in Fig. 3. The transmitted symbol s(t) in fm modulation scheme using orthogonal functions can be derived from Fig. 2 and expressed by (65):

$$s(t) = \sum_{i=1}^{m} x_i \sqrt{E_i}\, g_i(t) \quad (65)$$

In the above expression we assume that x_i is -1 if the bit is zero and +1 if the bit is one, $\{g_i\}$ is the set of m orthonormal functions and $\{E_i\}$ is the energy of the orthonormal signal for bit i. Using the above expression we can show that the BER probability is given by (66):

$$P_{B_i} = Q\left(\sqrt{\frac{2E_i}{N_0}}\right) \quad (66)$$

The above result shows that the BER for the orthogonal fm receiver is same as that of the BPSK scheme. It is well known [20] that BPSK gives the lowest possible BER in all of the existing communication schemes.

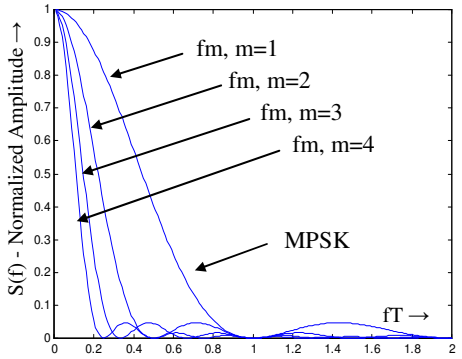

Figure 11. PSD comparison

V. CONSTRAINED GRAM SCHMIDT

It should be realized by now that the function modulation method requires a systematic approach for generating wave forms suitable for the concepts presented here. We use the words waveforms, functions, bit functions, and symbols interchangeably.

One of the major constraints that all waveforms must satisfy is the band limited property. The band limited property requires that the functions cannot have any discontinuity or sharp edges during the symbol period and also at the inter-symbol interfaces. The functions should be analytic if possible, that is, they will have smooth and continuous derivatives of all order. The orthogonality is another important requirement for the design of a simpler fm system. Otherwise all functions must be independent over the interval [0,T].

In this section, we describe a very general method, called Constrained Gram Schmidt (CGS) method, for generating band-limited orthonormal functions over [0,T]. We qualify the method as constrained because in addition to orthogonality, the functions satisfy many constraints, appropriate for digital communication. The method can be used for constrained independent functions also.

There are many methods of generating orthogonal functions. The methods in [21] cannot produce orthogonal functions with any kind of constraints on the nature of the resulting functions. A method for generating band-limited orthogonal functions, which are orthogonal over the finite symbol time interval, has been presented in [18]. However, that method also does not allow us to control the characteristics of the orthogonal functions it produces. Reference [26] describes a modem implementation method using the orthogonal functions of [18]. Chang [25] gives a method of generating band-limited orthogonal functions, which are orthogonal over infinite time interval. Again his method also does not allow us to use constraints on the functions. In the following paragraphs, we briefly describe the CGS method. There are many possible variations of CGS, which are not explored in this paper.

Let C^1 denote the class of all real valued continuous functions, with continuous first order derivatives, defined over the finite symbol time interval [0,T]. Let $F(t) = \{f_i(t) \in C^1,\ t \in [0,T]\}$ be a set of linearly independent functions with inner product defined by (67):

$$(f_i, f_j) = \int_0^T f_i(t) f_j(t) dt \quad (67)$$

The Gram Schmidt Orthogonalization (GSO) method as described in [10] is given by the equations in (68)

$$g_1(t) = f_1(t)$$
$$g_n(t) = f_n(t) + a_1 f_1(t) + a_2 f_2(t) + \cdots + a_{n-1} f_{n-1}(t),\ n \geq 2 \quad (68)$$

where the set of coefficients $\{a_i\}$ is obtained from the solution of the linear simultaneous equations (69):

$$b = Fa \tag{69}$$

Here b is a n-1 column vector with i-th element equal to the inner product (g_n , f_i), a is the n-1 column vector of unknown coefficients [a_1, a_2, … a_{n-1}]', and F is a square matrix of size n-1 with ij-th element given by the inner product (f_i , f_j). Our constrained approach extends the method defined by (69).

Communication channels require many different types of constraints on the symbols. Some of these requirements are stated below. The symbols (i) must join smoothly at inter-symbol boundary points, (ii) must not introduce any dc bias, and (iii) remain frequency band-limited. In addition the fm system may also need to use (iv) orthonormal functions.

Our objective is to generate a set of orthonormal functions $G(t) = \{g_i(t) \in C^1,\ i = 1 \dots m, t \in [0,T]\}$ from the set F(t) that will satisfy the above and similar other requirements. It should be realized that it is not necessary that the elements of the function space be constrained to start with. During the process of transmission they can be dynamically adjusted to satisfy the above constraints in real time. However, the present formulation and the methodology are sufficient to keep the symbols constrained.

We express the bit functions as a linear combination of sinusoidal functions as shown in (70). Expressions (70) will ensure that the bit functions in G(t) are analytic within the symbol interval [0,T].

$$g_i(t) = \sum_{j=1}^{M_c} c_{ij}\, Sin\left(w_{ij}t + \varphi_{ij}\right)\ t \in [0,T] \tag{70}$$

In equation (70) {w_{ij},} and {ϕ_{ij}} are some arbitrary and convenient choices for generating the functions. {w_{ij}} must be within the channel bandwidth making sure that G(t) is a band limited set. Each one of the functions in {$g_i(t)$} are created using a different set of frequency parameters. Since each set of sine functions for each $g_i(t)$ are independent, their linear combinations are also independent making G(t) an independent set. The value of M_c will depend on the number of constraints defined below by (71-75).

The constant coefficients {c_{ij}} of the linear ombination in (70) are selected to satisfy a series of constraints. We only mention some of the constraints that appeared to be necessary for the proper operation of the fm system as defined in this paper. Different communication channel may require different set of constraints. However the general concept presented here still covers many possibilities. The functions selected in (70) are also not the only choices. Any set of band limited and independent functions can be used to start with.

To synchronize the symbols at the receiver we make the symbols start and end at zero value (70). A sufficient condition for that is to make the bit functions behave the same way. Thus we consider the constraints (71) on G(t):

$$g_i(0) = 0, \quad g_i(T) = 0,\ i = 1 \dots m \tag{71}$$

To make the symbols join smoothly we want to impose the following derivative constraints (72) at the two ends of the bit functions.

$$\frac{d}{dt}g_i(t)|_{t=0} = \alpha, \qquad \frac{d}{dt}g_i(t)|_{t=T} = \alpha,\ i = 1 \dots m \tag{72}$$

In (15) α is any real number. In this paper we will set $\alpha=0$ merely for convenience. It is clear that if we want further smoothness at the symbol interfaces we can force the higher order derivatives to similar constraints. The constraints (71) and (72) will ensure that the entire symbol stream given by (58) is analytic. They also prevent any kind of discrete variations at the inter-symbol interfaces.

In many situations it may be necessary to avoid biasing the communication channel by a Direct Current (DC) voltage. To implement that requirement we set the integrals (73) of all bit functions to zero:

$$\int_0^T g_i(t)dt = 0,\ i = 1 \dots m \tag{73}$$

To be able to detect the symbols properly at the receiver we may need to make the symbols pass through some predefined points. We call them way points. This property (74) may also help to synchronize the symbols properly.

$$g_i(t_k) = a_k,\ t_k \in [0,T],\ k = 1..K_i,\ i = 1..m \tag{74}$$

Here, {K_i} denotes the number of way points for the i-th bit function and {a_k } are some known choices.

If we want to generate orthogonal functions as bit functions then we include the constraints (75):

$$\int_0^T g_i(t)g_j(t)dt = 0 \quad i,j = 1..m, \quad i \neq j \tag{75}$$

Summarizing, the method for generating the bit functions is to substitute the expression for the bit function (70) into all the constraints (71-75) defined above. This substitution will produce several linear equations, similar to (69), for the set of unknown constants $\{c_{ij}\}$. This set of simultaneous equations can then be solved for the constants. These constant coefficients will then be substituted back in (70) to get the analytical expression for each bit function. Note that for each bit function we have to solve a different set of equations like (69). The above process generates G(t) as an independent set of analytic functions with specified bandwidth.

We have used the constraints (71-74) to generate four non-orthogonal bit functions. That is, we did not use the orthogonality constraints defined by (75). These bit functions are shown in Fig. 6 and Fig. 7. By including the constraint (75) we have created constrained orthogonal functions shown in Fig. 12.

Constrained Gram Schmidt (CGS) is a very powerful method of constructing orthogonal functions. The original Gram Schmidt Orthogonalization (GSO) algorithm is very

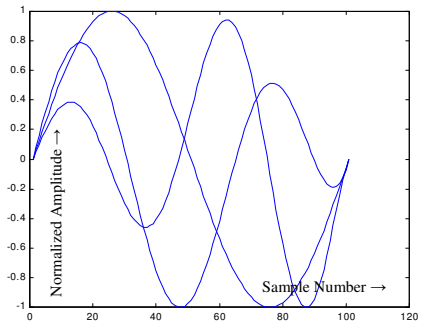

Figure 12. Constrained Orthogonal Functions

well known in literature. It is widely used for many communication problems, like Global Positioning Systems. However GSO was not enough for generating orthogonal functions with specific characteristics. We have extended GSO to CGS where you can create orthogonal functions with many additional properties. CGS will be very helpful in fm system for creating band limited orthogonal and non-orthogonal functions with desired properties.

VI. THE CAPACITY THEOREM

In this section we show that the function modulation method has higher capacity than the SHK methods. We also extend the Shannon's capacity theorem to a new higher capacity result and show that the fm method can be used to achieve such a capacity.

A. *Infinity Assumption*

Shannon presented his capacity theorem almost sixty years back. Lot of research has been done on this subject since then. During the early phase most of the focus was on finding alternatives [12] of sinc functions for the reconstruction of original function from the sample values. During the later phase it seems that the focus got shifted to dimensionality [27] aspect of the theorem. It appears that people have [28] assumed that T is constant and finite, which is not true. Shannon said in his paper [2] many times that T will go to infinite value in the limit. No one, it seems, has ever paid any attention to finite time issue of the engineering requirements. Recent research [29] has found that under certain assumptions ultra narrow band systems can produce capacity higher than that predicted Shannon. However the majority of research [30][31] work is now focused on comparing the performance of their systems using Shannon's theorem as a measure.

In this section we go back to the original theorem [2] and take a look at one of its core assumptions, the infinite time assumption, of the symbol duration. This subject was raised because we were sampling our symbols at a very high rate in our software radio approach. Apparently this violated the sampling theorem and the dimensionality theorem. To prove the general engineering practice of high sample rate we looked into the original theory, which eventually lead to the capacity theorem. Both capacity and sampling theorems were presented in the same paper [2] by Shannon.

In the next few paragraphs we show how Shannon [2] used infinite time in the derivation of his capacity theorem. He used m to denote number of bits to be transmitted and M to represent the number of symbols. These two are related by the well known equation (76):

$$M = 2^m \tag{76}$$

This relationship is very important in digital communication engineering. As before, we point out that we do not transmit m bits, we transmit M symbols. From these M symbols we find out how many bits they represent using the above relation (76). Thus the focus of capacity theorem is not on the bits but on the symbols. How many symbols our receiver can detect is the main concern in the derivation of the capacity theorem.

After that, Shannon defines the capacity C using the symbol time T by (77):

$$C = \lim_{T\to\infty} \frac{log_2 M}{T} \tag{77}$$

Thus clearly, his assumption is that T is very large and is going to go to infinity eventually. He writes near the above definition "…M different signal functions of duration T on a channel, ..". In another place he repeats, "There are 2^m such sequences, and each corresponds to a particular signal function of duration T". Thus T is the symbol time and he assumes all his symbols are of infinite time durations.

Shannon's entire theory, derivation, and proof depend on this infinite time assumption. He writes "The transmitted signal will lie very close to the surface of the sphere of radius $\sqrt{2TWP}$, since in a high-dimensional sphere nearly all the volume is very close to the surface". Here P is the average signal power and W is the channel bandwidth. He achieves high dimension by assuming T as very large. Thus his proof will be invalid if we assume that the symbol time T is finite and small.

An interesting observation can be derived as a result of his infinite time assumption. He writes "The quantity TW log(1+P/N) is, for large T , the number of bits that can be transmitted in time T." This means that the bits cannot be recovered until the time T ends. Therefore bits/sec is not really meaningful here. Bits are not coming out of the system every second. He confirms "There will be, therefore, an overall delay of 2T seconds". Thus actually the receiver stops when T approaches infinity.

The total number of bits per symbol is infinity in Shannon's case. Because T is infinity in the expression for bits, as mentioned in the previous paragraph, TW log(1+P/N). It is true that any finite rate over infinite time will also give infinite bits. But we have shown that it is not happening here. Thus Shannon's result indicates – infinite capacity. We show that we have the same conclusion even over finite time interval.

A well explored statement in [2] is the following: "Then we can say that any function limited to the bandwidth W and the time interval T can be specified by giving 2WT numbers." The number WT is actually infinity, because T is infinity. In the literature however, it has been presented as if it is a finite number [7, p. 93]. This finite interpretation to Shannon's proof has generated a large volume of research papers similar to [12]. We have shown that the above finiteness interpretation violates a very well known and a fundamental mathematical theory that says all functions are infinite dimensional vectors even over finite and small time intervals and therefore cannot truly be represented by finitely many numbers. In this section we show that this infinite time assumption is not necessary.

B. The (P+N)/N Factor

Now we examine how Shannon [2] got the expression (P+N)/N in his capacity theorem. Here N is the average noise power, averaged over the symbol time T. The received signal power is P+N. The concept used in our geometric approach is actually deeply embedded in the geometric approach of [2].

Consider the ASK scenario shown in Fig. 13. The allowed amplitude levels are shown by two dashed lines. For every dashed line there is a band over which the amplitude can swing because of the noise in the channel. This band is shown by the continuous lines with width proportional to $\sqrt{N}$. The total number of symbols M, i.e. number of amplitudes, which can be transmitted, is then given by (78):

$$M = \frac{Interval\ Height}{Band\ Height} = \frac{OA}{BA} \tag{78}$$

Since the amplitude is proportional to the square root of power, the above expression reduces to (79):

$$M = \sqrt{\frac{P+N}{N}} \tag{79}$$

This is the maximum limit we can achieve, at this frequency for an ASK system. In a later section we consider all possible frequencies. In Fig. 13 we used: (A) Discrete bounds and (B) Sinusoidal functions. If we relax these two SHK conditions then we can significantly improve the capacity expression (79).

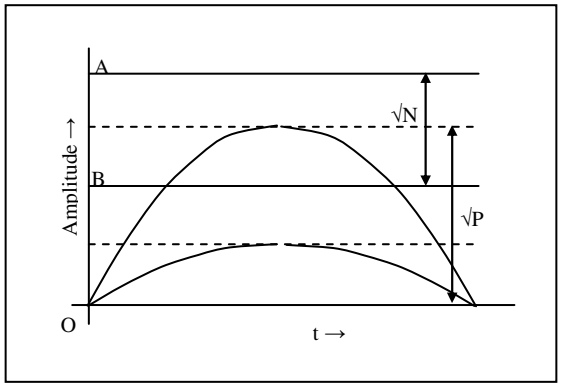

Figure 13. Discrete approach

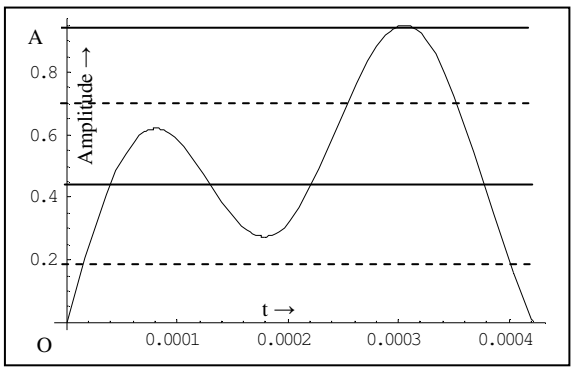

Figure 14. Analog approach

Consider Fig. 14 now, where we have relaxed both conditions mentioned above. In Fig. 14 we have shown only one general function. But you can imagine many such functions going up and down over the entire amplitude interval OA and crossing all bands many times. Fig. 15 shows many such allowed functions. Virtually there is no limit of the number of symbols or functions that can be transmitted or plotted and that can be distinguished also. These are the kind of functions fm uses. Thus higher capacity can be achieved by using the fm communication method as opposed to the discrete or the SHK methods.

Shannon has used such general class of functions in his derivation of capacity theorem. He wrote – "Actually, two signals can be reliably distinguished if they differ by only a small amount, provided this difference is sustained over a long period of time".

From the above statement we can see that the noise bands can be modified to a new format as shown in Fig. 16. In this figure we show two fm symbols along with their noise bands or pipes around them. One important difference between the noise bands in fm and SHK methods is that in fm the noise band is dynamic and moves with the function and not static, discrete, or straight lines like in Fig. 13.

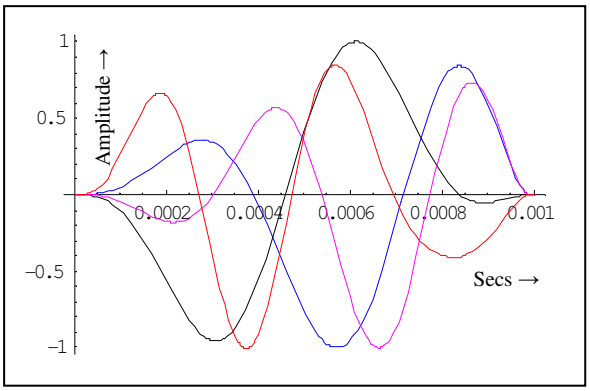

Figure 15. Bit-Functions for fm method

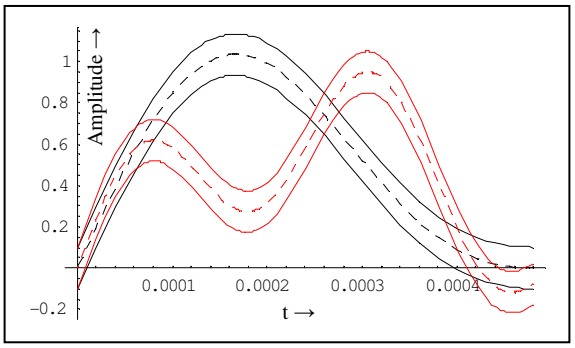

Figure 16. Noise bands

These fm bands are like transparent and flexible pipes around the functions. In fm one symbol can penetrate the noise band of another symbol. They can remain together for some duration and then get separated to distinguish themselves. This fm band is the minimum distance between two symbols. As long as the distance between two symbols is greater than this minimal distance, over only a small interval of time, the symbols will remain detectable. The important fact is that this concept of overlapping bands or flexible pipes around a function is not present in the SHK communication systems. This fact significantly increases the number of allowable symbols in the fm system.

Thus the capacity of fm method is much higher than the SHK methods and we show later that SHK cannot achieve Shannon's limit but fm can.

C. The WT Factor

Now consider the signals used in Fig. 17 and ask the Shannon's question [2] – "How many different signals can be distinguished at the receiving point in spite of the perturbations due to noise?" In this section we derive the same answer that he got but with the assumption that T is finite and small.

The approach is to configure the function space into small discrete rectangles, as shown in Fig. 18, and count all possible symbols that can pass through these rectangles. We

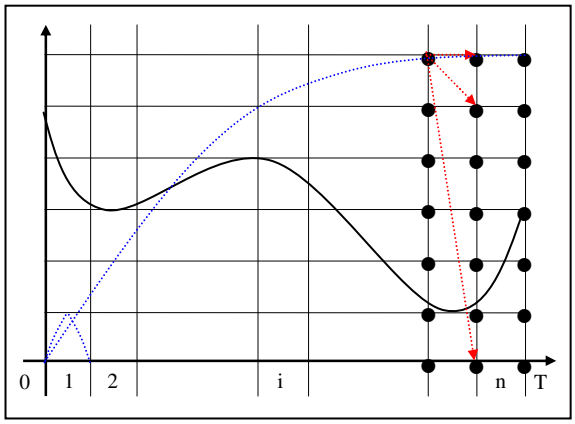

Figure 17. Nyquist sample intervals

divide the interval [0, T] on the x-axis into n equal parts. Where n is the number of samples and is equal to 2WT, because we are using the Nyquist sampling rate in this figure. The signal amplitude interval on the y-axis is $\sqrt{(P+N)}$ and is broken in to q number of y-subintervals, where $q = \sqrt{(P+N)/N}$. Thus the two dimensional plane becomes a grid of rectangular boxes. Each box can be considered to have a point through which only one detectable function can pass. Some of these points are shown, only in the last two vertical columns, to avoid too much clustering. The continuous line shows an example of a symbol function, which can pass through these dots.

We can connect any dot in one vertical column to any dot in the next or previous vertical column to create a portion of a function passing through these rectangles. The arrows in the figure show some such possible connections that the functions can take. These connection lines will not violate the bandwidth limit of the function, because the length of a t-subinterval is equal to the Nyquist length. Thus between two vertical columns there can be qxq=q^2 number of functions. Counting this way we can see that the total number of symbols, M, in the entire grid of Fig. 17, can be expressed by (80):

$$M = \left(\frac{\sqrt{P+N}}{N}\right)\left(\frac{\sqrt{P+N}}{N}\right)\ldots\left(\frac{\sqrt{P+N}}{N}\right) \tag{80}$$

Since it has 2WT terms the above simplifies to (81):

$$M = \left(\frac{\sqrt{P+N}}{\sqrt{N}}\right)^{2WT} \tag{81}$$

This is the same factor in the capacity formula [2]. This construction process can be used to generate detectable band limited functions for the fm method proving that the fm can achieve the Shannon's limit.

We have shown two sine functions using dashed lines in Fig. 18. One of them has the highest frequency and lowest detectable amplitude and the other one has the lowest frequency but highest possible amplitude. It is easy to see that if we use only ASK design then we can get qWT number of symbols. This is because for each frequency we get q number of amplitudes and there are WT numbers of full cycle sine functions possible over 2WT sample intervals. Although it is not known if there are any ASK system that uses these frequencies all at a time. We also see from this grid design and the graphs drawn that Fourier sine functions will not be able to cover all the grids the way general functions can. This gives a geometric proof that sinusoidal approach cannot achieve the Shannon's capacity results.

In this subsection we provided a proof of Shannon's theorem that did not require infinite time interval assumption. However in a small symbol time T the Nyquist rate will give very few samples and we will not be able to

recover a symbol as shown by examples in Fig. 1. Next we show how a higher sample rate enables us to detect more symbols thus increasing the capacity.

D. The Higher Sampling Rate

Assume as before that T is small and finite and we sample at k times the Nyquist rate, where k > 1. Therefore n on the x-axis of Fig. 17 is now equal to 2kWT. Since each t-subinterval is very small now, the noise energy due to N is also very small on these sub-intervals and therefore the equivalent noise band will be proportional to $\sqrt{N}/k$ on these subintervals as shown in the Fig. 18. However, the range of total signal variation still remains $\sqrt{(P+N)}$ over the entire symbol time T. Thus the total number of y-axis intervals is $\sqrt{(P+N)}/(\sqrt{N}/k)$ and is equal to kq. So the grid is k times finer in both t and y axes.

In this finer grid any point on the vertical line for any t-subinterval cannot be connected to any point on the vertical line on the next or previous t-subinterval, because that will make the function rise much faster and violate the bandwidth condition. It is easy to understand, however, that a point in one t-subinterval can be connected to only q consecutive number of points in the next or previous t-subinterval without violating the bandwidth constraint.

If we consider any consecutive q rows then we can see that the situation is very similar to the Fig. 17. Since there are 2kWT numbers of columns, then the number of functions generated by any q horizontal block can be expressed by (82):

$$M = \left(\frac{\sqrt{P+N}}{\sqrt{N}}\right)^{2kWT} \tag{82}$$

This value when converted into bits/sec, using the formula (77) for capacity, will reduce to (83):

$$C > \frac{log_2 M}{T} = kW log_2\left(\frac{P+N}{N}\right) \tag{83}$$

We have used greater than notation because we did not count all the functions in this finer grid. Although it is possible to count all possible functions and get a very precise expression instead of (83). But that counting result may not lead to the well known and simple expression like (83). We are only interested in showing that the capacity depends on the sample rate and therefore theoretically can go to infinity. Expression (83) is only a lower bound. We can now state the following new capacity theorem:

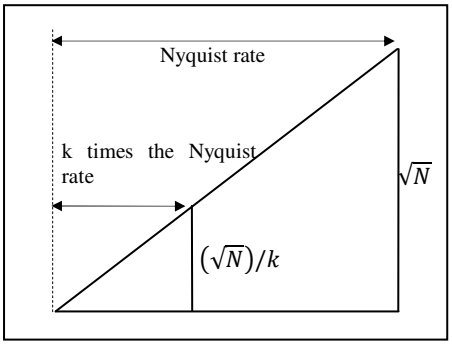

Figure 18. Noise at higher sample rate

Theorem 5: Assume that any arbitrary set {P, N, W, T, m} is given. Where P=signal power, N=channel noise power, W=channel bandwidth, T=finite symbol time, and m=number of bits per symbol. Then there exists a fm system, with m continuous and independent bit functions of the given bandwidth W, and a sample rate that is k times the Nyquist rate, with k > 1, which will satisfy the relation

$$m = kWT\ log_2\left(\frac{P+N}{N}\right) \tag{84}$$

Also, this resultant fm system will output m bits in every T seconds.

The statement of the above Theorem 5 is in many ways different from the original [2] statement. Most importantly it explicitly mentions the symbol time T in the statement. It should be noted that T is not infinity in our case. It is believed that the lack of such a mention of T in the original statement of Shannon created a lot of confusion in the literature. The Theorem 5 statement also describes a digital communication system for achieving such a capacity which was missing so far in the communication theory. Finally the statement essentially points out that the capacity is dependent on the technology, the sample rate.

We can see that the essential idea behind higher capacity is very similar to creating a high resolution camera. Higher the number of pixel in a digital camera better is the picture quality. In the communication case we are essentially resolving the two dimensional plane of the function space into a very high resolution grid by sampling at a very high rate.

Thus high rate sampling is equivalent to selecting a high resolution camera. This camera will allow us to see all the details of a function and thus giving the ability to detect more symbols. The result (83) is quite obvious and is expected also. Essentially we have reduced the noise by a factor of k, the sample rate, which thus produced higher capacity. In this context we should examine the definition of noise. It is actually dependent on the signal processing technology and the algorithms we use in our receiver. Thus in this sense our result is nothing new. If you can reduce the noise you can increase the capacity, which was obvious from the original Shannon's theorem. All we did is we brought out this noise factor k outside the capacity expression. However, we have also given a method, the fm method, for implementing the Shannon's theory, which was

missing so far in the literature. And finally of course the finite duration solution.

It is possible to come to the same conclusion (83), however, from another direction also. Slepian has shown [18] that there are infinitely many band limited orthogonal functions over a finite interval of time. These Slepian functions can be used in the fm method of Fig. 2 to transmit theoretically infinite number of bits. This has been described using the limit operation of the finite term Fourier series (56). We have also shown that it is not necessary to use orthonormal functions to get the same results. The series (56) can be valid for independent functions also.

Thinking philosophically, we can ask is the capacity really limited. If we look through our windows then we can see the nature outside, the blue sky, patches of clouds, mountains, trees, and plants. We can close and open our eyes and see the same view instantly. This is because our eyes and the brain working together as a receiver is immensely advanced and powerful. We have evolved over billions of years to this state of our mind, body, and soul. Thus the communication capacity of the air medium channel between the nature outside and our eyes is infinity. Assuming that is the definition of infinity, of course. The fm system presented here uses our state of the art technology. As our technology evolves we will get higher and higher capacity. The technology behind the digital camera is an example of one such step toward higher capacity. Thus it will be wrong to think that the capacity of a digital communication channel is limited and is independent of our technology.

E. Discussion

In this paper our focus is on the sampling theorem and the capacity theorem over finite duration signals. The main question we asked is how many samples we need for a finite duration signals. Our objective is not very much on how to reconstruct a finite duration signal from its samples, rather how many samples are necessary to correctly reconstruct a finite duration signal. We found that more you sample better will be your reconstruction. Or in the other words the original Nyquist rate is not good enough for recovery of signals of finite duration.

We approached the capacity theorem from the same angle. Original theory assumed that the symbols must be of infinite duration. We wanted to see what happens when symbols are of finite duration. To perform this analysis we used the finite duration sampling theorem.

Shannon's approach defines capacity as the number of symbols that a receiver can distinguish. We have also used the same concept in this paper. Thus the ability to distinguish symbols is the key issue of capacity. Clearly this then depends on what kind of symbols you are using in your system.

If we use general purpose independent functions then the ability to detect and distinguish will also depend on the computational power of the receiver. More sophisticated the algorithm is more will be the demand on the processor power. However the demand on the ADC is not very important. We have shown that ADC does not have to be very fast. Internal sampling by interpolation can be very effectively used to increase the resolution. We have also discussed a global approach to signal processing. Thus computational power is not of immediate concern.

If we use orthogonal functions then our fm theory shows that computational burden is very low. All we have to do is to increase the number of parallel integrators in our Application Specific Integrated Circuit (ASIC). More symbols mean more integrators. Observe that in this orthogonal case, we have simplified the Shannon's approach by introducing the concept of bit functions. In orthogonal case we do not work on the symbol space but in the bit functions space which requires significantly lower number of integrators. This approach shows why capacity is higher in the orthogonal fm systems.

Because our present technology provides powerful processors, we now have very high computational capability. As a result we can revisit low bandwidth channels, like POTS, to provide high capacity data rate. The fm scheme and the algorithm presented here, based on software radio and global approach, essentially leads to that kind of direction. It is not always necessary to require high bandwidth channel to provide high capacity data rate. This fm theory can be used with all the existing theories. For example any compression algorithm can be used in conjunction with fm system to further enhance the performance. It may be possible to use many of the existing symbols in a fm scheme. It is also quite feasible to use fm symbols over a sinusoidal carrier.

The fm scheme is basically an analog approach for digital communication. All symbols in this scheme are just like continuous analog functions. No discrete concept is embedded in the symbol or in the symbol stream. Only transmitter and receivers are digital. Thus we are taking full advantage of the nature which is analog. We are not impinging any discrete disturbances in the analog world.

VII. THE SPHERE PACKING

Shannon represents [2] every symbol as a point in an n-dimensional Cartesian space. He used the entire set of Nyquist samples of a symbol as coordinates in this space. The total number of samples of a symbol is WT and increases as T increase. Thus the dimension of his space eventually increases to infinity. Since the power in each symbol is fixed all the symbols lie on the surface of a sphere of constant radius. This is because the sum of the square of the sample values is the power and is also is a measure of the distance from the origin. Thus every symbol is a point on the surface of the same sphere. Because of noise these symbols will become like a non-overlapping billiard ball centered on these points [32][33, pp. 655-659]. In this section we show another geometric representation, also in Cartesian

space, of a function using the infinite dimensionality concept of function space.

In Fig. 19 we show a function f(t) taken from C[a,b] space, the space of continuous functions. We consider the interval [a,b] as finite and small in Fig. 19. Fig. 20 shows the corresponding representation of the function in a real n-dimensional space. Here n is any finite and fixed number, not necessarily large and is not going to infinity, its value can be two also.

We can partition the interval [a,b] in many smaller subintervals as shown by marks in Fig. 19. Each such small subinterval can be sampled n times and can be represented by one point in the n-dimensional space, the same way Shannon did. Thus the interval [a, t1] of Fig. 19 is represented by the point t1 in Fig. 20. There is no shortage of points in the function shown in Fig. 19. We have shown that infinite sample rate is meaningful because the function is infinite dimensional over any finite interval. The smaller the intervals are, larger will be the number of points in Fig. 20. If we join these points by a smooth line then we get the dashed line, which represents the function, as shown in Fig. 20.

We have also shown, by solid lines, how the noise band or pipe around the function can be represented in the same way in the n-dimensional space of Fig. 20. As mentioned, in Fig. 20 these pipes are now the symbols in the n-dimensional space. All these pipes, in the n-dimensional space, are flexible, transparent, and one pipe can penetrate or join another pipe for a period and then get separated. Contrary to sphere packing case of Shannon, where the spheres cannot overlap, in Fig. 20 we do not have that restriction. We can see now that the end points can indeed overlap and the functions will still be detectable. This flexibility of the pipes makes it possible to pack infinite number of pipes in the n-dimensional sphere. Thus proving that the capacity can be indeed infinity even when we use this n-dimensional geometric concept.

Fig. 19 and Fig. 20 are in some sense identical. Both are line graphs, one in two dimensional function space and the other one is in n-dimensional Cartesian space. Therefore it is really not necessary to go to Fig. 20 to analyze functions. It may be possible to analyze all aspects of a function using

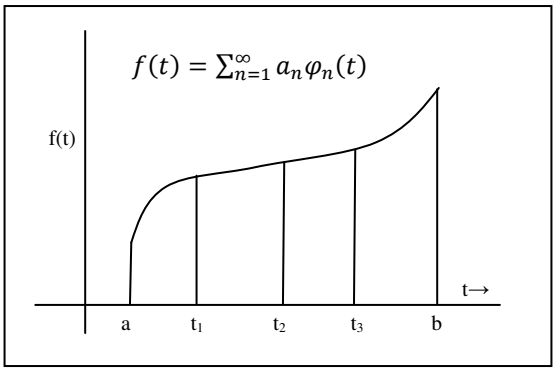

Figure 19. A function in C[a,b]

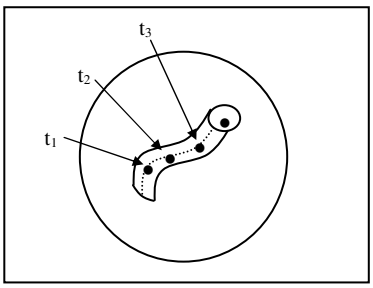

Figure 20. n-Dimensional view of a function from C[a,b]

Fig. 19. In some sense Fig. 19 has more information than Fig. 20. Fig. 20 represents only the coefficients of the infinite series or the sample values of the function. It does not contain the actual graph between the samples nor the orthogonal functions of the infinite Fourier series as in Fig. 19. We also point out again that the underlying concept used in this explanation is still based on infinite dimensionality property of function space.

VIII. CONCLUSION AND FUTURE WORK

In this paper we have presented an analysis of digital communication engineering using the theory of infinite dimensionality of function space. We have used non-sinusoidal symbols, finite duration theory, very high sample rate, and software radio approach to create a function modulation (fm) method. It is shown that this fm method can give much higher capacity than predicted by the original Shannon's theorem thus helping us to create a green modem.

The practical implementation of a high speed modem, based on the fm concept may require a large number of independent band limited functions. An in-depth research in that direction on a new powerful hardware is the next required milestone.

ACKNOWLEDGMENT

The first author is in many ways grateful for many suggestions and discussions with Pamela Das during the laboratory testing of the hardware implementation that helped to solve many problems. He is also deeply indebted to Hari Sankar Basak for his very careful and thorough review of the draft manuscripts at various stages of the research. It will be inappropriate not to acknowledge that the idea behind the Fig. 17 came from David Dyrnaes, a friend and colleague, during many discussions on the practical implementation of the fm concept.

REFERENCES

[1] S. Das, N. Mohanty, and A. Singh, "Capacity theorem for finite duration symbols", Porc. of IARIA International conference on networks, ICN 2009, Guadalupe, France, published by IEEE Press, 2009.

[2] C. E .Shannon, "Communication in the presence of noise", Proc. IEEE, Vol. 86, No. 2, pp. 447-457, 1998.

[3] S. Das, N. Mohanty, and A. Singh, "Is the Nyquist rate enough?", Proc. of IARIA Internaitoanl confercen on digital telecommunicaitons, ICDT 2008, Bucharest, Romania, published by IEEE Press, 2008.

[4] P. A. Laplante, *Real-time systems design and analysis,* Third Edition, IEEE Press, New Jersey, 2004.

[5] Y. Eideman, V. Milman, and A. Tsolomitis, *Functional analysis, an introduction*, Amer. Math. Soc, Providence, Rhode Island, 2004.

[6] E. M. Vestrup, *The theory of measures and integration*, Wiley, New Jersey, 2003.

[7] G. M. Phillips, *Interpolation and approximation by polynomials*, Can.Math.Soc., Springer, New York, 2003.

[8] V. K. Ingle and J. G. Proakis, *Digital signal processing using Matlab*, Brooks/Cole, California, 2000.

[9] T. M. Cover and J. A. Thomas, *Elements of information theory*, Second Edition, John Wiley, New Jersey, 2006.

[10] R. Bellman, *Introduction to matrix analysis*, SIAM, Philadelphia, 1995.

[11] N. Mohanty, *Signal processing*, Van Nostrand, NY,1987.

[12] L. W. Couch II, *Digital and analog communication systems*, Second Edition, Macmillan, NY, USA, 1987.

[13] A. J. Jerri, "The Shannon sampling theorem, its various extensions and applications – a tutorial review", Proc. IEEE, Vol. 65, No. 11, 1977.

[14] S. Das, N. Mohanty, and A. Singh, "A function modulation method for digital communications", Wireless telecommunications symposium, WTS 2007, Pomona, California, USA, published by IEEE Press, 2007.

[15] J. Farlow, J. E. Hall, J. M. McDill, and B. H. West, *Differential equations linear algebra*, Prentice Hall, New Jersey, 2002**.**

[16] C. C. Pugh, *Real mathematical analysis,* Springer, NY, 2002

[17] J. Gibson, *The communication handbook*, CRC Press, Chapter 68, 2002.

[18] D. Slepian, Some comments on Fourier analysis, uncertainty and modeling, SIAM Review, Vol. 25, pp. 378-393, Jul 1983.

[19] S. Das and N. Mohanty, "A narrow band OFDM", IEEE vehicular technology conference, VTC Fall 2004, Los Angeles, California, USA, published by IEEE Press, 2004.

[20] B. Sklar, *Digital communications fundamentals and applications,* Prentice Hall, NJ, USA, 1988.

[21] G. H. Golub and C. F. Van Loan, *Matrix computation*, Third Edition, Johns Hopkins university press, pp. 236-264, 1996.

[22] TMS320C5402 Hardware board, Texas instruments, Dallas, Texas, USA, 2001.

[23] TMS320C5000 Code composer studio, Integrated development environment, V2.0, Texas instruments, Dallas, Texas, **USA,** 2001.

[24] Fuqin Xiong, *Digital modulation techniques*, Artech House, Boston, USA, 2000.

[25] R. W. Chang, "Synthesis of band-limited orthogonal signals for multichannel data transmission," Bell Syst. Tech. J., Vol 45, pp. 1775-1796, December 1966.

[26] V. V. Loginov and V. A. Kochanov, "Implementing the high speed modem with multidimensional modulation using MS32c542 DSP," Spra321, Texas Instrument, September 1996.

[27] D. Slepian, "On bandwidth", Proc. IEEE, Vol. 64, No. 3, pp. 379-393, March 1976.

[28] A. D. Wyner and S. Shamai, "Introduction to communication in the presence of noise by C. E. Shannon", Proc. IEEE Vol.86, No.2, pp. 442-446, 1998.

[29] F. Man and W. Lenan, "Extension to Shannon's channel capacity – The Experimental Verification",Proc. ISISPCS,Nov 2007, China, published by IEEE Press, 2007.

[30] P. Stocia,Y. Jiang, and J. Li, "On MIMO channel capacity: an intuitive discussion",IEEE, Sign.Proc.Mag., May 2005.

[31] Jun Wang, Shi-hua Zhu, and Lei Wang, "On the channel capacity of MJMO-OFDM systems",Proc. ISCIT 2005, published by IEEE Press, 2005.

[32] T. D. Schneider, "Claude Shannon: biologist", IEEE Eng. Med. Bio. Mag., Feb. 2006.

[33] B. P. Lathi, *Modern digital and analog communication systems*, Third edition, Oxford University Press, 1998.