Application of Soft Computing in Coherent Communications Phase Synchronization

Jeffrey T. Drake[1] and Nadipuram R. Prasad

The Klipsch School of Electrical and Computer Engineering MSC 3-0
New Mexico State University
P.O. Box 30001
Las Cruces, NM USA, 88003-8001
E-mail: jeff.drake@gsfc.nasa.gov

The use of soft computing techniques in coherent communications phase synchronization provides an alternative to analytical or hard computing methods. This paper discusses a novel use of Adaptive Neuro-Fuzzy Inference Systems (ANFIS) for phase synchronization in coherent communications systems utilizing Multiple Phase Shift Keying (M-PSK) modulation. A brief overview of the M-PSK digital communications bandpass modulation technique is presented and it's requisite need for phase synchronization is discussed. We briefly describe the hybrid platform developed by Jang [6] that incorporates fuzzy/neural structures namely the, Adaptive Neuro-Fuzzy Interference Systems (ANFIS). We then discuss application of ANFIS to phase estimation for M-PSK. The modeling of both **explicit**, and **implicit** phase estimation schemes for M-PSK symbols with unknown structure are discussed. Performance results from simulation of the above scheme is presented.

1. INTRODUCTION

The functional block diagram shown in Figure 1 illustrates the signal flow through a simplified digital communications system. The upper blocks–format and modulate indicate the signal transformations from the source to the transmitter. The lower blocks indicate the signal transformations from the receiver to the sink; the lower blocks essentially reverse the signal processing steps performed by the upper blocks. The channel in our work is a source of complex additive white Gaussian (AWGN) noise.

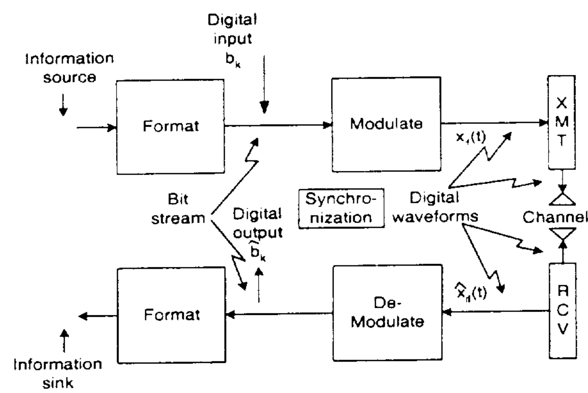


Figure 1. Simplified digital communications system

In the case of coherent phase modulation also know as Phase Shift Keying (PSK), the receiver is assumed to be able to generate a set of reference signals whose phases are identical to the phases of the signaling alphabet in use at the transmitter. These reference signals are compared to the incoming signals in the process of making *maximum likelihood* symbol decisions. In order to be able to generate these reference signals, the receiver has to be in synchronization with the received carrier. This means that there has to be phase concurrence between the incoming carrier sinusoid and a replica of it in the receiver. In other words, if there were no information modulated on the incoming carrier, the incoming sinusoid and the replica in the receiver would pass through zero simultaneously. This is what is known as being in *phase lock* and it is a condition that must be closely approximated if coherently modulated signals are to be accurately demodulated at the receiver. It is in this context that we use the terms *synchronization* and *estimation* somewhat interchangeably. We further illustrate these concepts in the remainder of this section before moving on to our

[1] Corresponding author affiliation and address: J. Drake is also with the Goddard Space Flight Center, National Aeronautics and Space Administration, Code 567, Greenbelt, MD 20771 USA

novel application of soft computing to synchronization.

Figure 1 indicates that from the source to the modulator a *message* such as the ASCII character "E", is converted in the *format* block to a *baseband signal* or *bit stream*. This bit stream is characterized by a sequence of digital symbols. These digital symbols are uniformly space pulses representing the message. After modulation, the message takes the form of of a digitally encoded waveform or *digital waveform*.

We focus on PSK modulation in this work. PSK modulation is now widely used in both military and commercial communications systems. The general analytic expression for PSK is

$$x_{if}(t) = \sqrt{\frac{2E}{T}} cos[\omega_o t + \theta_i(t)] \quad 0 \le t \le T \qquad (1)$$

where the phase term (*baseband signaling alphabet*), $\theta_i(t)$, will have M discrete values, typically given by

$$\theta_i(t) = \frac{2\pi i}{M} \quad i = 1, ..., M$$

The *bandpass signaling alphabet* (simply the phasors generated by the PSK modulator) are then

$$a_i = e^{j\theta_i(t)} \quad i = 1, ..., M$$

For the binary PSK (BPSK) example in Figure 2, *M* is 2. The parameter *E* is the symbol energy, *T* is the symbol time duration, and $0 \le t \le T$. In BPSK modulation, the modulating data shifts the phase of the waveform, $x_{if}(t)$, to one of two states, either zero or π (180^o). The concepts of information, baseband representation, baseband signaling alphabet, bandpass signaling alphabet and bandpass digital waveforms are illustrated in Figure 2 which shows the ASCII binary representation of the letter "E" and the resulting BPSK waveform. We note the rapid phase changes at the symbol transitions. We see that the information is carried in the phase of the sinusoidal carrier wave. For M=4 or quadrature phase shift keying (QPSK) the modulator maps 2 baseband bits of the bit stream to one of 4 possible phases (phasors).

The *demodulator* in Figure 1 has the task making the best *estimate* $\hat{a}_i$ of the transmitted *bandpass signaling alphabet* a_i. The *optimum* receiver in this case is know as a *correlation* receiver [18]. The demodulation process requires multiplying the received waveform $x_{if}(t)$ by a *reference* waveforms $r_i(t)$ which have frequency and phase *identical* to that of the unmodulated sinusoid used to transmit the waveform (set M=1 and i=1 in (1)) and sampling a *matched filter* at the optimum sample time $\hat{T}$. Figure 3 illustrates the correlation receiver where we have separated the functions of frequency, phase and symbol timing synchronization and we use *complex* signal notation

$$e^{j\alpha} = cos(\alpha) + j\ sin(\alpha) \qquad (2)$$

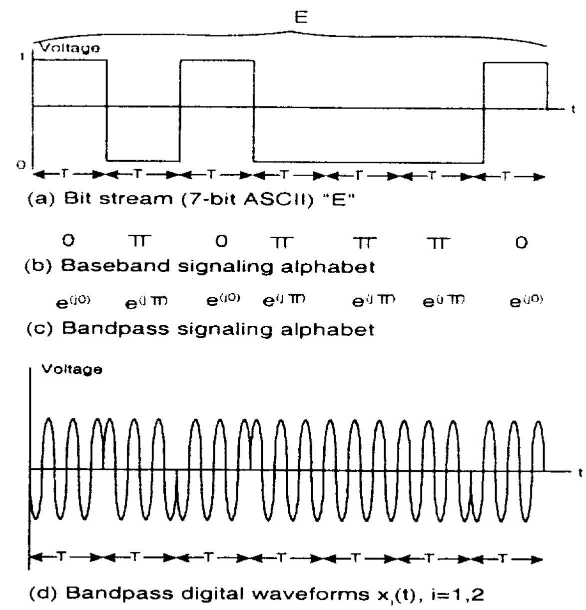


Figure 2. Bit stream and corresponding digital waveform for Binary-PSK (BPSK)

which allows for easier generalization to M>2 and makes for a more compact mathematical notation.

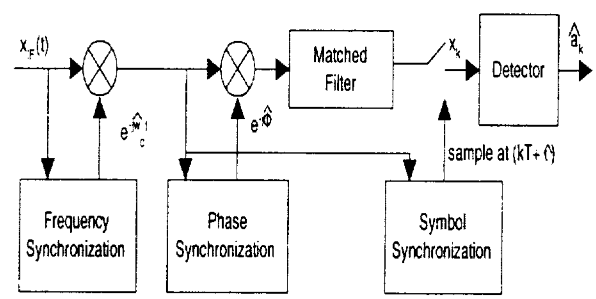


Figure 3. Coherent communications receiver

Recapping, synchronization (estimation) is a critical function in any modern coherent digital communications system. In synchronous digital transmissions the information is conveyed by uniformly spaced pulses, and the received signal is completely known except for the data symbols and a group of variables referred to as *reference parameters*. Reference parameters in this context include ω, ϕ, and τ the carrier angular frequency, carrier phase and symbol timing, respectively. Though it is the ultimate task of the receiver to generate an accurate replica of the symbol sequence with no regard to the reference parameters, this is only possible by exploiting knowledge of these parameters. Coherent demodulation is used with passband digital communications. In coherent communications the baseband data signal is derived making use of a local reference with the same frequency and phase as the incoming carrier. Carrier or phase synchronization is the function of aligning the phase and frequency of the receiver oscillator with that of the transmitter oscillator when the information is modulated onto the carrier.

The coherent receiver structure that forms the basis for our work is shown in Figure 4. We make the assumption that the frequency and symbol timing are know exactly. This assumption is valid because generally these reference parameters can be estimated independently of each other. Referring to Figure 4, for carrier phase estimation we represent the received signal by the sufficient statistic, namely,

$$x(k) = a_k e^{j\phi} + n_k \tag{3}$$

Here, a_k is the possibly complex valued bandpass signaling alphabet symbol $a_k \in \{e^{j\frac{2\pi i}{M}} : i \in \{0, 1, \cdots, M\}\}$, ϕ is the unknown carrier phase, and n_k is complex additive white Gaussian noise (AWGN). It is shown by Van Trees in [18] that $\{x(k)\}$ forms a set of sufficient statistics for estimating the phase ϕ. In simple terms a sufficient statistic indicates that no other information about the signal is need to obtain the "best" estimate of the parameter(s).

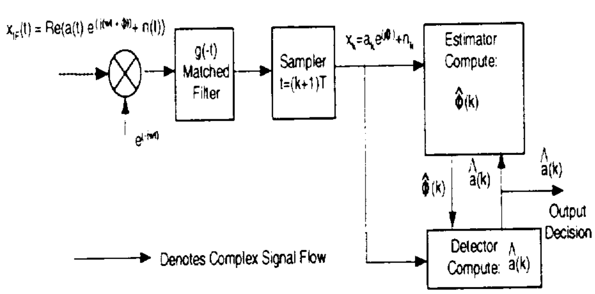


Figure 4. Coherent receiver structure

Historically the approaches to synchronization structure can be divided into two categories, which we denote as ad-hoc structures and derived or analytical structures. Arguably, the most commonly used analytical method for phase estimation is that of maximum likelihood (ML). The ML estimator has several important theoretical advantages which make it very desirable to obtain. These include its' being the *best* w.r.t the chosen criterion, the that it has the *possibility* of achieving the lower bound on performance know as the Cramer-Rao lower bound or CRB. An estimator that achieves the CRB is known as an efficient estimator and if an estimator is efficient it is a ML estimator. The ML estimate is given by the *likelihood equation*

$$arg_{\phi} max \ p(\mathbf{x}|\phi) \tag{4}$$

or by the equivalent expression

$$\left.\frac{\partial \ln p_{\mathbf{x}|\phi}(\mathbf{X}|\phi)}{\partial\phi}\right|_{\phi=\hat{\phi}_{ml}(x)} = 0 \tag{5}$$

Where $p(\mathbf{x}|\phi)$ the probability density function of the received vector $\mathbf{x}$, given ϕ. The ML estimate $\hat{\phi}$ can be thought of as the phase ϕ that *most likely* gave rise to the received signal $\mathbf{x}$.

Unfortunately, in most practical cases where digital modulation is present, derived structure criterion leads to highly non-linear systems, which in general cannot be solved for the optimum solution and only implicit solutions are arrived at. To find explicit solutions approximations must be made, which leads to results that are valid only for ranges of the parameters and are in essence sub-optimal to the true ML estimate.

Equation (6) gives the *exact* ML estimator of carrier phase for M-PSK modulation obtained in terms of the received signal over the immediate past N symbols. As pointed out by Kam [7] the ML phase estimator is nonimplementable. Implementable approximations by which $\hat{\phi}$ can be obtained have been made [7, 16, 12]. An important and fair question is: "Do better estimation procedures than ML exist?" The answer is that if an efficient estimate $\hat{\phi}$ does not exist then it is certainly possible. The difficulty is that no procedure exists for finding such an estimator other than trial and error. The highly nonlinear result for the exact ML estimate of $\hat{\phi}$ provides the necessary motivation to examine the application of soft computing methods as an alternative for phase estimator design. SC offers a model free approach to estimator design and provides a viable alternative estimator "design" tool.

$$\cos\,\hat{\phi}(k) \sum_{l=k-N}^{k-1} \frac{\sum_{L=1}^{M} e^{-S_L} \sinh q_L(l,\hat{\phi}(k)) \Im[x(l)a_L^*]}{\sum_{L=1}^{M} e^{-S_L} \cosh q_L(l,\hat{\phi}(k))} \tag{6}$$

$$= sin\,\hat{\phi}(k) \sum_{l=k-N}^{k-1} \frac{\sum_{L=1}^{M} e^{-S_L} \sinh q_L(l,\hat{\phi}(k)) \Re[x(l)a_L^*]}{\sum_{L=1}^{M} e^{-S_L} \cosh q_L(l,\hat{\phi}(k))} \tag{7}$$

Where $S_L = \frac{|a_L|^2}{N_0}$, $q_L(l,\phi) = \frac{2}{N_0}\Re(x(l)a_L^* e^{-j\phi})$, and c a constant independent of ϕ. N is the block size in symbols.

2. Soft Computing and Estimation

"As the complexity of a system increases, our ability to make precise and yet significant statements about its behavior diminishes until a threshold is reached beyond which precision and significance (or relevance) become almost mutually exclusive characteristics."

Lotfi Zadeh
Professor, Systems Engineering, 1973

Zadeh's "Principle of Incompatibility", quoted above [22], accurately describes the the situation we encounter with ML estimation of the phase in M-PSK. Thus we look for alternative estimation algorithm design methodologies. Soft computing is one such alternative. SC techniques enable construction of estimation models using only target system samples or stated another way, SC offers a model-free design approach.

Our aim here is not to necessarily derive a carrier phase recovery scheme for M-PSK that offers improvements in performance or simplification in implementation compared to existing methods, although these goal are believed attainable. Rather, the main contribution lies in developing new model-free tools for the design of estimation schemes and greater insights into carrier phase estimation [3].

Many problems in estimation and identification can be formulated as function-approximation problems [9]. For instance, in conventional system identification, input-output data is gathered from a physical system and a least-squares approach can be used to provide the best approximation for the linear function that maps the system inputs to its outputs. In a similar fashion, in parameter estimation if the given data is such that it associates measurable system variables with an internal system parameter, a functional mapping may be constructed that approximates the process of estimation of the internal system parameter. A system which exhibits *universal approximation* is capable of approximating any real continuous function on a compact set to any degree of accuracy. Certain classes of fuzzy systems have the property of *universal approximation* [14, 9]. The proof of fuzzy systems as *universal approximations* was first given in [20].

We briefly describe Jang's [6] hybrid platform that incorporates fuzzy/neural structures namely the, Adaptive Neuro-Fuzzy Interference Systems (ANFIS). We then discuss application of ANFIS to phase estimation for M-PSK.

We first discuss application of ANFIS to the general case of **explicit** (open-loop) phase estimation for blocks of N, M-PSK symbols. We find that this most general estimation scheme is difficult to implement in practice.

We then engage ANFIS in identification of a *novel* **implicit** (closed-loop or error tracking) estimator of phase for continuous transmission or large blocks of M-PSK modulation using no a-priori knowledge of the signal or estimator structure. We discuss the results and performance of this *new* estimation scheme.

3. An Brief Review of Neuro-Fuzzy Integration: ANFIS

In this section we provide a brief description of Jang's [6] hybrid platform that incorporates fuzzy/neural structures namely the, Adaptive Network Fuzzy Inference System (ANFIS). We limit the review of ANFIS to that required to discuss the material presented within this paper. ANFIS is a *universal approximator* and as such is capable of approximating any real continuous function on a compact set to any degree of accuracy [6]. Thus in parameter estimation where the given data is such that it associates measurable system variables with an internal system parameter, a functional mapping may be constructed by ANFIS that approximates the process of estimation of the internal system parameter.

ANFIS refers to a class of adaptive network-based fuzzy inference systems which are functionally equivalent to fuzzy inference systems [6]. Specifically the ANFIS system of interest here is functionally equivalent to the Sugeno first-order fuzzy model. We briefly review Jang's [6] *Hybrid Learning Algorithm*, which combines gradient descent and the least-squares method, and discuss how the equivalent fuzzy inference system can be rapidly trained and adapted with this algorithm.

As a simple example we assume a fuzzy inference system with two inputs x and y and one output z. The first-order Sugeno fuzzy model with two fuzzy If-Then rules is the following:

Rule 1: If x is A_1 and y is B_1, Then $f_1 = p_1x + q_1y + r_1$,
Rule 2: If x is A_2 and y is B_2, Then $f_2 = p_2x + q_2y + r_2$,

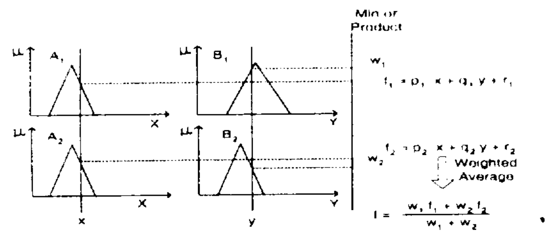
The resulting Sugeno fuzzy reasoning system is shown in Figure 5(a). Here the output z is the weighted average of the individual rules outputs and is itself a crisp value. The corresponding equivalent ANFIS architecture is shown in Figure 5(b). Nodes at the same layer have similar functions. We describe the node functions next. The output of the ith node in layer l is denoted as $O_{l,i}$.

Layer 1 Every node i in this layer is an adaptive node. Parameters in this layer are referred to as **premise parameters**.

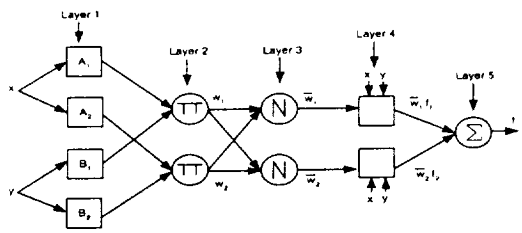
Layer 2 Every node in this layer is a fixed node labeled $\prod$, whose output is the product of all the incoming signals. Here each node represents the firing strength of a rule. Other T-norm operators that perform the fuzzy AND can be used as the node function at this layer.

Layer 3 All nodes in this layer are fixed nodes labeled N. The outputs of this layer are called **normalized firing strengths**.

Layer 4 All nodes i in this layer are adaptive nodes The parameters of this mode are called **consequent parameters**.



(a) Two-input first-order Sugeno fuzzy model with two rules.



(b) Equivalent ANFIS architecture.

Figure 5. Sugeno and equivalent ANFIS architecture.

Layer 5 This node is a fixed node labeled $\sum$, that computes the overall output as the summation of all incoming signals.

Through the definitions of layers 1-5 of the ANFIS structure we have an adaptive network-based fuzzy inference system which is functionally equivalent to Sugeno first-order fuzzy inference systems.

As an adaptive system the outputs of the adaptive nodes depend on the modifiable parameters of the adaptive nodes. The **learning rule** specifies how these parameters should be updated to minimize a prescribed error measure E. The error measure is a mathematical expression that measures the difference between the networks actual output and the desired output, such as the *squared error*. The basic learning rule of the adaptive network is the *steepest descent* method. In this method the gradient vector is derived by repeated application of the chain rule. Having obtained the gradient, if we use it in a steepest descent method, the resulting learning algorithm is called the **backpropagation learning rule**[2, 21, 15, 17, 6].

While the backpropagation learning rule can be used to identify the parameters in an adaptive network, this method is slow to converge. The **Hybrid Learning Algorithm** [6], which combines backpropagation and the least-squares method (LSE) can be used to rapidly trained and adapt the equivalent fuzzy inference system. For hybrid learning applied in batch mode (off-line learning), each epoch is composed of a **forward pass** and a **backward pass** as summarized in Table 3.

Table 1. Hybrid learning procedure for ANFIS

	Forward Pass	Backward Pass
Premise parameters	Fixed	Gradient descent
Consequent parameters	Least-squares est.	Fixed
Signals	Node outputs	Error signals

In the forward pass of the hybrid learning algorithm, node outputs go forward until the final layer (layer 4 in Figure 5(b)) and the consequent parameters are identified by the least-squares method. In the backward pass, the error signals propagate backward and the premise parameters are updated by gradient descent.

4. General Structure Identification for ANFIS

We introduced Jang's [6] ANFIS architecture along with its rules for learning in the previous section. The rules for learning only deal with **parameter identification**. Methods for **structure identification** to determine an initial ANFIS architecture are required before parameters can be tuned using the rules for learning. This process is equally important to the successful application of ANFIS.

Structure identification involves the following issues:

- Selection of relevant input variables
- Initial ANFIS architecture
  1. Input space partitioning
  2. Number and type of MFs for each input
  3. Number of fuzzy if-then rules
  4. Premise and consequence parts of rules
- Select initial MF parameters

In a conventional fuzzy inference system, the number of rules is determined by an expert who is familiar with the target system to be modeled. If one has insight into the system at hand and the system is not too complicated then structure identification can easily be done by a human expert. This is typically the Mamdani method of developing fuzzy If-Then rules. The result is a Fuzzy Associative Memory (FAM) that provides the knowledge base for the system. With more complex systems or those where expert rules are not readily available other means are needed to identify the initial ANFIS structure. In such cases where no expert is available, the number of MFs assigned to each input variable is often chosen by visually inspecting the data sets or simply by trail and error. This situation is also common to neural networks where there is no simple way to determine in advance the minimal number of hidden units required to achieve a desired performance level [6]. In our applications we use simple *grid partitioning*. In grid partitioning the number of MFs on each input variable uniquely determines the number of rules [2] The initial values of the premise parameters are set such that the centers of the MFs are equally spaced along the range of of each input variable.

Efficient partitioning of the input space can decrease the number of rules and thus increase the speed of learning and adaptation. There are several techniques that have been used successfully in neural network structure identification [10]. One of the most effective structure identification methods for ANFIS is a binary tree partition scheme based on the CART (classification and regression tree) algorithm [1]. The CART algorithm was first applied to structure determination in fuzzy modeling by Jang in [5]. In the CART tree partition, each region can be uniquely specified along a binary decision tree. Tree partitioning eliminates the problem of exponential growth in rules that we see with grid partitioning. But often more MFs are needed per input to define the fuzzy regions, and these MFs do not generally bear clear linguistic meanings.

5. Explicit (open-loop) ANFIS Phase Estimation

Referring to Figure 4 and (3) we see that the observed (received) signal consists of signal plus noise. The signal is composed of transmitted *bandpass alphabet symbols* a_k that are multiplied (phase rotated) by an unknown phasor, $e^{j\phi}$, which represents the unknown phase offset between the receiver carrier and transmitter carrier sinusoid. The noise n_k is previously defined, complex AWGN. The unknown phase ϕ is treated as constant over the duration of observation of N observations. That is, all received signals have the same, unknown phase offset ϕ, which we must estimate.

The **ideal** ANFIS M-PSK explicit phase estimation scheme would be such that a functional mapping may be constructed that approximates the process of estimation of the internal system parameter. We illustrate this "idealized" system in Figure 6.

We would like to employ ANFIS to develop a model for an explicit (open-loop) estimator of phase using no a-priori knowledge of the signal and limited intuition from experts, training the system with the observed (received) signal and the target signal. The observed signal is as described at the

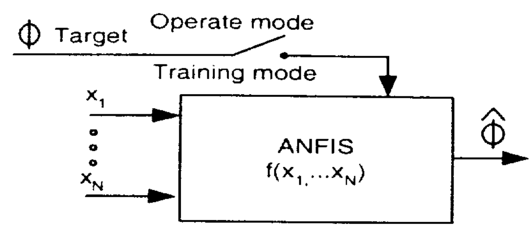


Figure 6. Ideal explicit ANFIS estimation

start of this section i.e. composed of transmitted *bandpass alphabet symbols* a_k that are multiplied (phase rotated) by an unknown phasor, $e^{j\phi}$ plus complex AWGN. Each packet or block is composed of N such observed signals. The target signal is the phasor, $e^{j\phi}$, which is constant over all N observations.

Due the the *curse of dimensionality* previously described, such an ideal ANFIS structure is not readily implementable. With improved structure or system identification methods such as CART (Classification and Regression Trees) [1] the ideal system may become a reality in the future. That is it may be possible to estimate the phase accurately if all N symbols could be presented to the ANFIS in parallel, an "adequate" number of MFs could be used on each input and the ANFIS trained. But at present the explosion of rules severely limits the number of inputs if many MFs are required.

6. Implicit (closed-loop) ANFIS Phase Detector

In this section we employ ANFIS in identification of a *novel*, implementable, **implicit** (closed-loop or error tracking) estimator of phase for continuous transmission or large blocks of M-PSK modulation using no a-priori knowledge of the signal or estimator structure. We discuss the model, results and performance of this *new* estimation scheme.

A phase error tracking system (Figure 7) is characterized by the following principles of operation. A phase error signal as a function of the phase alignment error is computed in a functional block called a phase error detector. This error signal is then used in a feedback loop to adjust the voltage controlled oscillator (VCO). If properly designed, the feedback circuit forces the error signal to zero. The VCO is then aligned with the received signal and may serve as the reference phase in the receiver. For unmodulated carrier (CW) the phase error detector simply becomes a multiplier [4]. For modulated signals the structure of the phase error detector is much more complex. Many different phase error detectors have been proposed. They include the M-PSK Costas loop [4], *generalized* Costas loop [12], the tanlock loop [11], the Mth power loop [16] and the approximate

[2] There will be $(\#MFs)^{(\#inputs)}$ rules, this relationship is know as the "Curse of Dimensionality."

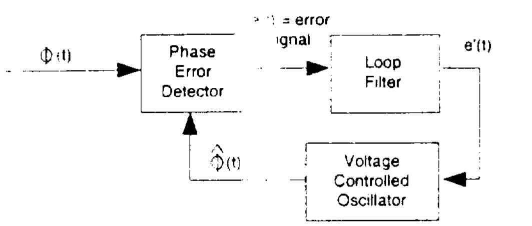


Figure 7. Basic phase error tracking system

ML loop [12, 8].

We apply ANFIS to identify a phase error detector for use with M-PSK and examine its performance in comparison to the the commonly used decision directed (DD), *modified* M-PSK Costas loop [12] (*High SNR Approximate ML loop [8]*).

Assuming *no* knowledge of the information symbols, the ANFIS phase estimator implementation is *not* decision-directed in nature. That is the estimation scheme does not not or utilize estimates of the transmitted symbols. In a fashion similar to that of the previous section, in this chapter we employ ANFIS to develop a model for a closed-loop estimator of phase of M-PSK modulation.

Recall from (3) the received signal or observations are complex valued. Attempts to model the phase detector using both magnitude and phase or both the in-phase and quadrature components of the observed signal are problematic. With either of these approaches one or both of the inputs has an unbounded domain. From (3) we see that due to noise the magnitude of the observed signal can take on any value from $[0\ \infty]$, as can the individual in-phase and quadrature components of the observed signal. This makes the task of partitioning the input space with MFs and training the ANFIS difficult.

As a first approach to skirt this problem of unlimited input domain size, we perform a rectangular to polar transformation on the observed signal but here use *only* the phase of the observed signal. We justify this in two ways: first the observation from Viterbi and Viterbi [19] that fixing the magnitude of the received signal to a constant value gave good performance, and secondly reducing the input dimension to one greatly reduces the number of fuzzy rules required to model the input. Moreover by transforming the observed signal into phase angles we limit the input domain to the interval $[-\pi\ \pi)$. By limiting the input domain we are able to partition the input space by the grid partition method and adequately cover the input space with a finite number of MFs. Other approachs under investigation by the author include use of CART to partition the input space and training the ANFIS at low SNRs with a limiter to restrict the range of the observations. Both methods shows promise.

The ANFIS system is trained with the *phase* of the observed signal and the target signal previously described for the explicit estimation scheme. Figure 8 shows the block diagram of the configuration used in training the closed-loop ANFIS. Having trained the ANFIS to model the phase error detectors for M-PSK we use the ANFIS phase error detectors in a second order, type II PLL [4] to estimate phase for M-PSK as shown in Figure 9 with loop parameters $\zeta = 0.85$, $\frac{B_L}{SR} = 0.19\%$.

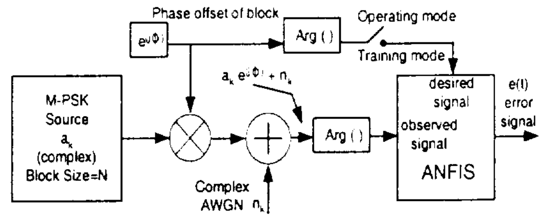


Figure 8. Closed-loop ANFIS training configuration

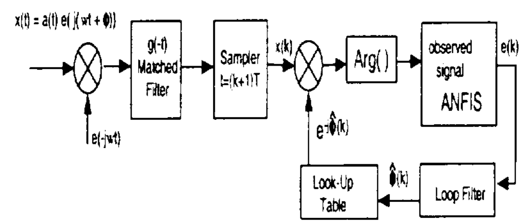


Figure 9. ANFIS M-PSK phase error tracking system

The purpose of the phase detector is to generate an output to function as a measure of the phase error [13]. A key tool to investigating phase acquisition and tracking is the average phase detector output commonly called the PD characteristic [3]. This is the expected value of the error signal $e(k)$, conditioned on a fixed value of the phase error $\phi_e \equiv \phi - \hat{\phi}$, i.e.,

$$S(\phi_e) \equiv E\left[e(k)|\phi_e\right] \tag{8}$$

From simulation $S(\phi_e)$ is obtained by opening the loop and measuring the average of the phase error signal. In Figure 10, we present the PD characteristics for 2-,4-,8, and 16-PSK obtained from the ANFIS based PD when the SNR=20dB. One detector cycle is plotted for each M, with the average detector output taken as the average of 1000 observations.

[3]The PD characteristic [13, 4] is also known as the S-curve [12]

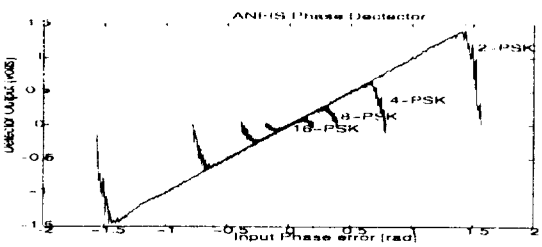


Figure 10. ANFIS PD characteristics for 2-,4-,8, and 16-PSK at SNR=20 dB

Having presented the resulting phase detector characteristics of the ANFIS M-PSK phase error detector we evaluate the variance of the error of ANFIS based, closed loop phase estimator. Recall that the variance is a measure power defined as

$$\sigma_{\hat{\phi}}^2 = E[\hat{\phi}^2] - E[\hat{\phi}]^2 \tag{9}$$

The variance of the estimation error provides an important measure of the estimator performance. We simulate the implicit (closed loop) ANFIS estimation scheme with Matlab® using the ANFIS derived phase error detector.

To determine the "optimal" number of membership functions required to achieve the minimum variance simulations where performed varying the number of MFs from 2 to 80. We plot results of these studies in Figure 11, where we see that "optimal" performance is achieved with about 20 MFs. Little or no gain is achieved by using greater than 20 MFs, as can be seen from Figure 11.

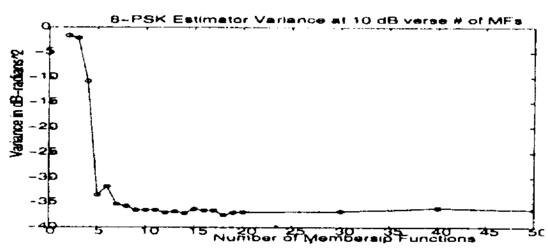


Figure 11. 8-PSK Estimator Variance at 10 dB verse Number of Membership Functions

To evaluate estimator performance we simulate the PLL described by the single set of loop design parameters given earlier in this section and with appropriate PDs (ANFIS and *Approximate ML*).

The variance data is presented for 8-PSK in Figure 12. The variance is estimated at each SNR as the average of 1000 observations of the steady state variance. Although not presented here, the estimator performance for 2,4 and 16 PSK was also evaluated and found to be identical to that of the corresponding *Approximate ML* estimators.

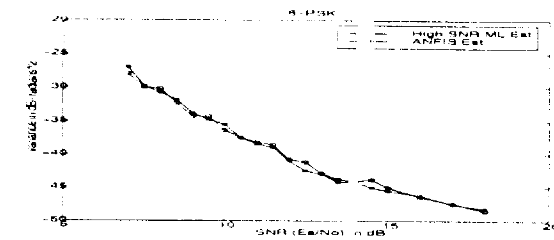


Figure 12. 8-PSK variance: ANFIS and High-SNR approximate ML Loop

We see that using only angle information to designing phase error detectors for M-PSK using the ANFIS model free approach, results in performance equivalent to that of the approximate ML solution. It is believed improved performance can be achieved by the ANFIS solution when the complete complex signal can be processed. This leads back to the explosion-of-rules that we have discussed previously as well as methods for restricting or limiting the range of noisy observations, while not reducing the information about phase contained in those observations.

7. Conclusions

We first described application of ANFIS to the general case of **explicit** (open-loop or feed-forward) phase estimation for blocks of N, M-PSK symbols. We saw that the "ideal" ANFIS phase estimation system is currently not implementable due to the explosion of rules. It may be possible to estimate the phase of a block of N symbols accurately if all N symbols could be presented to the ANFIS in parallel, an "adequate" number of MFs could be used on each input and the ANFIS trained But at present the explosion of rules severely limits the number of inputs if many MFs are required. Use of efficient input partitioning schemes such as CART [1] show promise in this area.

We then engaged ANFIS in identification of a *novel* **implicit** (closed-loop or error tracking) estimator of phase for continuous transmission or large blocks of M-PSK modulation using no a-priori knowledge of the signal or estimator structure. We discussed the results and performance of this *new* estimation scheme. We found that using only angle information in training phase error detectors for M-PSK using the ANFIS model free approach, results in performance equivalent to that of the approximate ML solution. The ANFIS techniques successfully applied to the design of M-PSK PDs is directly applicable to more general and complex Quadrature Amplitude Modulation PD design.

In the application of ANFIS to phase estimation the problem of *structure identification* remains a critical issue. It is believed improved performance can be achieved by the ANFIS solution when the complete complex signal can be processed. This leads back to the explosion-of-rules that we have discussed previously as well as methods for restricting or limiting the range of noisy observations, while not reducing the information about phase contained in those observations. As improved methods for structure identification are developed and applied to ANFIS, the explosion-of-rules problem can be overcome and improved ANFIS estimator performance and expanded applications are expected.

While the focus of this research was been on the application of soft computing to improve the performance of coherent communications phase estimation, soft computing techniques offer an alternative to current hard computing techniques in all areas of communications parameter estimation and synchronization.

References

[1] J. H. Breiman, R. Olshen, and C. Stone. *Classification and regression tress.* Wadsworth, Inc., Belmont, California, 1984.

[2] A. Bryson and Y.-C. Ho. *Applied optimal control.* Blaisdell, New York, 1969.

[3] J. T. Drake. *Communications Phase Synchronization Using The Adaptive Network Fuzzy Inference System (ANFIS).* PhD thesis, New Mexico State University, Las Cruces, NM, May 2000.

[4] F. Gardner. *Phaselock Techniques.* John Wiley, New York, 1976.

[5] J.-S. Jang. Structure determination in fuzzy modeling: a fuzzy cart approach. In *Proceedings, IEEE International Conference on Fuzzy Systems, 1996,* Orlando, Florida, 1994.

[6] J.-S. Jang, C.-T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence.* Prentice Hall, Upper Saddle River, New Jersey, 1997.

[7] P. Y. Kam. Maximum likelihood carrier phase recovery for linear suppressed-carrier digital data modulations. *IEEE Trans. Comm. Theory,* 34(6):522–527, June 1986.

[8] B. Kopp. *An Analysis of Carrier Phase Jitter in an MPSK Receiver Utilizing MAP Estimation.* PhD thesis, New Mexico State University, Las Cruces, NM, May 1994.

[9] E. G. Laukonen and K. M. Passino. Training fuzzy systems to perform estimation and identification. *Engineering Applications of Artificial Intelligence,* 8(5):499–514, 1995.

[10] T.-C. Lee. *Structure level adaptation for artificial neural networks.* Kluwer Academic, New York, New York, 1991.

[11] A. Makarios and P. Farrel. Noise and false-lock performance of psk-tanlock loop. *IEEE Trans. Comm.,* COM-30:2277–2284, Oct. 1982.

[12] U. Mengali. *Synchronization Techniques for Digital Receivers.* Plenum Press, New York, 1997.

[13] H. Meyr and G. Ascheid. *Synchronization in Digital communications.* John Wiley, New York, 1990.

[14] H. T. Nguyen and E. A. Walker. *A First Course in Fuzzy Logic.* CRC Press, New York, 1997.

[15] D. Parker. Learning logic. invention report s81-64. Technical report, Office of Technology Licensing, Stanford University, October 1982.

[16] J. G. Proakis. *Digital Communications, 3rd Edition.* McGraw Hill, New York, 1995.

[17] D. Rumelhart, G. Hinton, and R. Williams. Learning internal representations by error propagation. *Parallel distributed processing: explorations in the microstructure of cognition,* 1:318–362, 1986.

[18] H. L. Van Trees. *Detection, Estimation, and Modulation Theory.* John Wiley & Sons, New York, 1968.

[19] A. J. Viterbi and A. M. Viterbi. Nonlinear estimation of psk-modulated carrier phase with application to burst digital transmission. *IEEE Trans. Info. Theory,* 29(4):543–551, July 1983.

[20] L.-X. Wang. Fuzzy systems are universal approximations. In *Proceedings, 1st IEEE International Conference on Fuzzy Systems,* pages 1163–1170, San Diego, California, March 1992.

[21] P. Werbos. *Beyond regression: New tools for prediction and analysis in the behavioral sciences.* PhD thesis, Harvard University, 1974.

[22] L. A. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man, and Cybernetics,* 3(1):28–44, January 1973.