Article:

Zhang, L and Burr, A G orcid.org/0000-0001-6435-3962 (2004) Iterative carrier phase recovery suited to turbo-coded systems. IEEE Transactions on Wireless Communications. pp. 2267-2276. ISSN 1536-1276

https://doi.org/10.1109/TWC.20004.837407

Iterative Carrier Phase Recovery Suited to Turbo-Coded Systems

Li Zhang and Alister G. Burr

***Abstract*—This paper examines the problem of carrier phase recovery in turbo-coded systems. We introduce a new concept of "*a priori* probability aided phase estimation", where the extrinsic information (log-likelihood ratio) obtained from turbo decoder is used to aid an iterative phase estimation process, which is based on a maximum-likelihood strategy. The phase estimator operates jointly with the turbo decoding rather than separately prior to the decoder as in traditional approaches. This technique provides reliable phase estimation with variance of estimation errors approaching the Cramer–Rao bound at very low signal-to-noise ratio and allows robust decoding with a wide range of phase errors. This paper addresses its application in turbo-coded binary phase-shift keying and quaternary phase-shift keying systems over the additive white Gaussian noise channel. The bit-error-rate performance is investigated and shows that the performance of this technique is very close to the optimally synchronised system and significantly outperforms the traditional non-data-aided method without using additional pilot symbols.**

***Index Terms*—*A priori* information, Cramer–Rao bound (CRB), extrinsic information, iterative decoding, log-likelihood function (LLF), MAP, maximum-likelihood (ML) estimation, turbo codes.**

I. Introduction

FUTURE communication systems will be increasingly called upon to provide higher data rates and higher power efficiency. These requirements create the necessity of using powerful error control coding schemes, such as turbo codes [1], which are well known for their impressive near-Shannon limit error correcting performance. A communication system can be made highly robust by the use of turbo codes in a hostile environment. However, the application of turbo codes tends to exacerbate the synchronization problem because its very effectiveness leads to a low operating signal-to-noise ratio (SNR) (such as 1–2 dB). At this level, a data-aided (DA) or decision-directed (DD) synchronization is preferred to non-data-aided (NDA) methods. These two methods, however, require either long preambles, which increase redundancy, or access to decoding decisions, which are generally not available until synchronization has been performed. Furthermore, turbo codes are rather sensitive against phase mismatch, an even moderate offset may cause severe degradation of system performance. Therefore, a proper technique is needed to preserve the remarkable performance of turbo codes in presence of imperfect synchronization.

A number of previous publications have looked into this, particularly for parallel concatenated convolutional codes (PCCCs). Risley *et al.* [2] and Wiberg [3] increased the robustness of the system against phase errors by taking into account the design of the encoder. Risley *et al.* [2] presents design methods for turbo trellis-coded modulation (TCM) over fading channels, which can achieve significant coding gain on channels with phase distortion. In [3], the turbo decoding is made robust against phase uncertainty by using nonsystematic, rotationally invariant component codes. Mielczarek and Svensson [4] improved phase estimation in turbo-coded system using an enhanced turbo decoder by including the signs of the phase offset as additional states and accordingly modifying the turbo decoder. However, this leads to greater complexity. Similarly, [5] forms a finite-state Markov model for the fading channel phase. The estimation of channel phase and data is done jointly, on the supertrellis, which merges the trellises of the code and phase model, or on the separate trellises, via the Forward–Backward algorithm. Publications [6]–[9] present methods using soft output generated by the turbo decoder to aid the phase recovery. Morlet *et al.* [6] presents a tentative decision directed carrier phase estimation technique, where tentative symbol decisions calculated by the Viterbi decoder are used to replace the data decision in the DD method. The approach proposed in [7] is specifically geared for turbo coding. It uses tentative decisions of the first soft in, soft out (SISO) decoder during the decoding process in the phase recovery system as the symbol reference. A joint turbo decoding and carrier phase recovery algorithm is presented in [8]. It exploits the power of the extrinsic information generated in the iterative maximum *a posteriori* (MAP) decoder as metrics to perform the carrier phase acquisition and tracking. These works have demonstrated how the use of tentative decisions improve the carrier phase recovery and obviously outperform techniques using hard decisions when a turbo code is employed. But they do not make full use of the special properties of the iterative turbo decoding algorithm. A more recent paper [9] improves this. The authors propose an iterative soft-decision directed (ISDD) carrier phase estimation algorithm based on the maximum-likelihood (ML) strategy suited for coherent detection of turbo-coded quadrature amplitude modulation (QAM) modulation. The soft decisions

Manuscript received December 17, 2002; revised July 1, 2003; accepted September 22, 2003. The editor coordinating the review of this paper and approving it for publication is C. Xiao. This work was funded by QinetiQ under the MoD applied research program. This paper was presented in part at the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications PIMRC'2001, San Diego, CA, September. 2001, and at the IEEE International Symposium on Information Theory 2002, Lausanne, Switzerland, July, 2002.

L. Zhang was with the Department of Electronics, University of York, York YO10 5DD, U.K. She is now with the School of Electronic and Electrical Engineering, University of Leeds, Leeds LS2 9JT, U.K. (e-mail: l.x.zhang@leeds.ac.uk).

A. G. Burr is with the Department of Electronics, University of York, York YO10 5DD, U.K. (e-mail: alister@ohm.york.ac.uk).

Digital Object Identifier 10.1109/TWC.2004.837407

provided by the SISO decoders are utilized in the ML phase estimation. However, the approximation of the log-likelihood function (LLF) made in the process of attaining the estimate may lead to degradation to the estimation accuracy. One of their results will be used to compare with ours.

This paper presents a new concept, which we will call "*a priori probability aided (APPA) phase estimation*", which fits particularly well with the iterative turbo decoder, although it would also be applicable with other types of code. It can be understood as a generalization of DA and NDA synchronization, since it reduces to the former when perfect data knowledge is available, and to the latter when there is no *a priori* knowledge of the data. In place of hard data decisions, as in the DD case, it uses soft information obtained from the SISO decoder, such as the MAP decoder used for turbo codes. The algorithm is implemented based on the ML strategy and makes use of the extrinsic information log-likelihood ratio (LLR) produced iteratively by a MAP decoder within the decoding loop. The phase estimator and the turbo decoder operate simultaneously once per decoding iteration. The new phase estimate and extrinsic information are then employed in the next iteration. Applied iteratively, this technique allows successive refinement of the carrier phase estimate, until the joint decoder/synchroniser converges on the correct data and phase estimate. The estimate is worked out directly from the LLF using a complexity-reduced algorithm to avoid introducing excessive delay to the system. This approach does not change the structure of the codes or the decoding algorithm, unlike [3] and [4]. It nevertheless improves the synchronization by making full use of the existing resource (the extrinsic information), without adding extra complexity to the system.

We address phase estimation in turbo-coded binary phase-shift keying (BPSK) and quaternary phase-shift keying (QPSK) systems in this paper and initially assume a perfectly known symbol timing (set as 0). The outline of this paper is as follows. The first section briefly introduces the turbo codes, particularly the iterative turbo decoder and the extrinsic information. In the second section, we derive the modified LLF and the simplified algorithm to attain the ML estimate for BPSK and QPSK systems. The third section evaluates the performance in terms of bit error rate BER, mean phase estimate, and the mean square estimation error (MSEE). Finally, the last section summarizes the paper.

II. Brief Introduction to Turbo Codes

The scope of this paper is restricted to the PCCCs. However, the proposed technique can be straightforwardly extended to other coding schemes, such as serial concatenation of convolutional codes, turbo-product codes, and low-density parity check (LDPC) codes, etc.

The turbo encoder is built using a parallel concatenation of two identical recursive systematic convolutional (RSC) codes with generators $[G_1, G_2]$ linked together by an interleaver with size N, where G_1 and G_2 are the polynomials of the feedback and output connectivities of the RSC encoders. Both RSC encoders inputs use the same information data bits but according to a different sequence due to the presence of the interleaver. The parity bits out of the two encoders are properly punctured to achieve the desired coding rate. The turbo decoder based on iterative decoding consists of two SISO component decoders corresponding to the two component encoders. The concept behind turbo decoding is to pass soft decisions from the output of one decoder to the input of the other, and to iterate this process to produce better decisions. The most widely known decoding algorithm is the BCJR-MAP algorithm [10]. A variant of this algorithm, the Log-MAP algorithm [11], which operates in the logarithmic domain, is also a good choice considering its simpler implementation and performance equivalent to true MAP. The decoder (either MAP or Log-MAP) computes the soft bits using the logarithm of *a posteriori* probability ratio (LAPPR) associated with each data bit d_i, defined as

$$\Lambda_L(d_i) = \log\left(\frac{P(d_i = 1|\mathbf{Y})}{P(d_i = 0|\mathbf{Y})}\right) \tag{1}$$

where $\mathbf{Y}$ is the received sequence, and $P(d_i = j|\mathbf{Y}), j = 0, 1$ is the *a posteriori* probability (APP) of data bit d_i. The decoder decides $d_i = 1$ if $P(d_i = 1|\mathbf{Y}) > P(d_i = 0|\mathbf{Y})$, i.e. $\Lambda_L(d_i) > 0$, and $d_i = 0$ otherwise.

From Bayes' rule [12], the LAPPR (1) can be written as

$$\Lambda_L(d_i) = \log\left(\frac{P(\mathbf{Y}|d_i = 1)}{P(\mathbf{Y}|d_i = 0)}\right) + \log\left(\frac{P(d_i = 1)}{P(d_i = 0)}\right) \tag{2}$$

with the second term representing *extrinsic* information, denoted as $L(i)$. This LLR value, or generally the estimation for each data bit is passed to the other component decoder and serves as *a priori* information in the next decoding step. In this way, each decoder takes advantage of the extrinsic information produced by the other decoder at the previous step. The absolute magnitude of $L(i)$ is the measure of the reliability of the estimation; the larger, the higher the probability of the data bit being 1, when it is positive, or 0, when it is negative. As iterative decoding proceeds, the reliability improves, until the decoding converges to the correct decision after some iterations.

Such a decoding scheme suggests a new means of synchronization by making use, in the phase estimation process as in the decoding, of this iteratively improved extrinsic information which is inherent in the decoding process, with no additional computation.

III. APPA Carrier Phase Estimation

The algorithm of phase estimation using the *a priori* information provided by the iterative turbo decoder is derived from ML estimation considerations. The algorithm is described in two steps: first, we derive the LLF for the carrier phase estimation through which we integrate the *a priori* information into the ML estimation, and second, we approximate the LLF in order to attain the ML phase estimate with low complexity.

A. Log-Likelihood Function

Express the ith received symbol as $r_i = A_i \exp(j\phi_i)$, where A_i denotes its amplitude and ϕ_i represents the argument; both vary due to the existence of the carrier phase error and the noise.

According to [13], assuming an additive white Gaussian noise (AWGN) channel with a noise power of σ^2, the likelihood function for the estimation of carrier phase ϕ based on the observation of N received signals is

$$\Lambda(\phi) = \exp\left[\frac{1}{\sigma^2}\sum_{i=1}^{N}\mathrm{Re}\left(r_i s_i^*(\phi)\right)\right] \tag{3}$$

where $s_i(\phi)$ is the ith transmitted signal, which is a function of the carrier phase ϕ. Here we consider ϕ to be unknown but constant over one block. For M-PSK modulation, the symbol values are

$$s(m,\phi) = \exp\left[j\left(\frac{2\pi m}{M} + \phi\right)\right], \quad m = 0\ldots M-1 \tag{4}$$

where M is the size of the constellation.

We average the likelihood function over all data values using statistics of the signal. Typically, a uniform distribution is used in conventional NDA method, when no *a priori* knowledge is available. However, in the turbo-coded system, the extrinsic information is available after the first decoding iteration, from which we can obtain the probabilities of each possible transmitted symbol given the received signal, represented as $P_i(m)$, meaning the probability that the ith transmitted signal is the mth constellation point. Averaging the likelihood function over the constellation using these probabilities, we get the likelihood function for the proposed method,

$$\Lambda(\phi) = \prod_{i=1}^{N}\left\{\sum_{m=0}^{M-1} P_i(m)\exp\left[\frac{\mathrm{Re}\left(r_i s^*(m,\phi)\right)}{\sigma^2}\right]\right\}. \tag{5}$$

Take the logarithm of (5), then the LLF turns out to be

$$\begin{aligned}\Lambda_L(\phi) &= \sum_{i=1}^{N}\log\left\{\sum_{m=0}^{M-1} P_i(m)\exp\left[\frac{\mathrm{Re}\left(r_i s^*(m,\phi)\right)}{\sigma^2}\right]\right\}\\ &= \sum_{i=1}^{N}\Lambda_L^i(\phi)\end{aligned} \tag{6}$$

where $\Lambda_L^i(\phi)$ is defined as LLF for each individual symbol

$$\Lambda_L^i(\phi) = \log\left\{\sum_{m=0}^{M-1} P_i(m)\exp\left[\frac{\mathrm{Re}\left(r_i s^*(m,\phi)\right)}{\sigma^2}\right]\right\} \tag{7}$$

and the overall LLF (6) can be looked upon as the summation of $\Lambda_L^i(\phi)$ over N symbols.

In the following, we derive the LLF in binary phase-shift keying (BPSK) and quaternary phase-shift keying (QPSK) modulation separately.

As given in (2), the extrinsic information $L(i)$ is a LLR defined as

$$L(i) = \log\left(\frac{P(d_i=1)}{P(d_i=0)}\right). \tag{8}$$

In binary transmission, $P(d_i=1)+P(d_i=0)=1$, and hence, we can compute the *a priori* probabilities [14] from $L(i)$ by

$$\begin{aligned}P(d_i=1) &= \frac{\exp\left(L(i)\right)}{1+\exp\left(L(i)\right)}\\ P(d_i=0) &= \frac{1}{1+\exp\left(L(i)\right)}.\end{aligned} \tag{9}$$

1) BPSK: In BPSK, there are two possible transmitted symbol values

$$\begin{aligned}s(0,\phi) &= -\exp(j\phi)\\ s(1,\phi) &= \exp(j\phi).\end{aligned} \tag{10}$$

Hence, $P_i(m)$ can be obtained straightforwardly from (9) with

$$\begin{aligned}P_i(0) =& (d_i=0)\\ P_i(1) =& (d_i=1).\end{aligned} \tag{11}$$

Substituting (10) and (11) into (7), after some algebra, we get the LLF of the i^{th} symbol for BPSK system

$$\begin{aligned}\Lambda_{L,\text{BPSK}}^i(\phi) = \log\Bigg\{&\mathrm{sech}\left(\frac{L(i)}{2}\right)\\ &\times\cosh\left[\frac{L(i)}{2} + \frac{A_i\cos(\phi-\phi_i)}{\sigma^2}\right]\Bigg\}.\end{aligned} \tag{12}$$

2) QPSK: In the case of QPSK, there are four possible signal states rather than two, thus, every QPSK symbol transmits two bits. The QPSK signal values are expressed as

$$\begin{aligned}s(m,\phi) &= \exp\left[j\left(\frac{2\pi m}{4} + \phi + \phi_0\right)\right]\\ m &= 0,1,2,3\end{aligned} \tag{13}$$

with ϕ_0 a constant (generally equal to $\pi/4$).

Because of the interleaver, it is reasonable to assume that the two bits of the mth QPSK symbol, which are designated as (u_m, c_m), are independent. And each bit takes the value of 0 or 1 according to the mapping rule.

We correspondingly signify the extrinsic information for these two bits as $L_u(i)$ and $L_c(i)$. Given $L_u(i)$ and $L_c(i)$, $P_i(u_m)$ and $P_i(c_m)$ can be easily obtained using (9). As the two bits are independent of each other, $P_i(m)$ for QPSK is

$$P_i(m) = P_i(u_m)P_i(c_m). \tag{14}$$

Inserting (13) and (14) into (7), after simplification, the LLF for the ith symbol of QPSK is

$$\begin{aligned}&\Lambda_{L,\text{QPSK}}^i(\phi)\\ &= \log\Bigg\{\frac{1}{2}\mathrm{sech}\left(\frac{L_u(i)}{2}\right)\mathrm{sech}\left(\frac{L_c(i)}{2}\right)\\ &\qquad\times\Bigg[\cosh\left(\frac{A_i\cos(\phi-\varphi_i)}{\sigma^2} - \frac{L_u(i)}{2} - \frac{L_c(i)}{2}\right)\\ &\qquad+\cosh\left(\frac{A_i\sin(\phi-\varphi_i)}{\sigma^2} + \frac{L_u(i)}{2} - \frac{L_c(i)}{2}\right)\Bigg]\Bigg\}\end{aligned} \tag{15}$$

where we define $\varphi_i = \phi_i - \phi_0$ for convenience.

Observing the LLF formula for BPSK (12) and QPSK (15), we found that the the extrinsic information $L(i)$ in BPSK, or $L_u(i)$ and $L_c(i)$ in QPSK, have been embedded into the LLF for the phase estimation. As mentioned before, the absolute value of the extrinsic information represents a sort of reliability metrics of the data estimation: the larger, the higher the probability of the data bit being 1, when it is positive; or 0, when it is negative. For example, for $L(i) = 4$, the probability of the data bit being 1, i.e., $P(d_i = 1)$ is 98.2%. That is, the decoder is very certain about the decision, and the APPA estimator resembles the DA method. On the other hand, at the first decoding iteration, when the extrinsic information is equal to 0, $P(d_i = 1) = P(d_i = 0) = 50\%$, which is exactly the same as in the NDA estimation. Although, strictly speaking, these values are not *a priori* information for the whole receiver, since they are not known before the symbol is received, we stick to this term because of its well-established use in turbo decoding and because the information is *a priori* as far as the synchroniser is concerned. Hence, also we use the term "*a priori* probability aided (APPA) estimation" for our new approach to carrier estimation.

B. Low-Complexity Maximum-Likelihood Estimation (MLE)

The ML estimate $\hat{\phi}_{ML}$ is the value of ϕ that maximizes the overall LLF (6). The necessary condition for a maximum is

$$\frac{d\Lambda_L(\phi)}{d\phi} = 0. \tag{16}$$

From the previous section, substituting (12) and (15) into (6), the overall LLF for both modulation methods turns out to be highly nonlinear and complicated. It involves too much calculation and would introduce processing delay if we were to compute the exact estimate directly from its derivative satisfying condition (16). To resolve this problem, we expand the LLF as a Fourier series

$$\Lambda_L(\phi) = \sum_{i=1}^{N}\left\{a_0^i + \sum_{n=1}^{\infty}\left[a_n^i \cos(n(\phi-\phi_i)) + b_n^i \sin(n(\phi-\phi_i))\right]\right\} \tag{17}$$

with Fourier coefficients defined as

$$\begin{aligned} a_n^i &= \frac{1}{\pi}\int_{-\pi}^{\pi} \Lambda_L^i(\phi)\cos(n\phi)d\phi \\ b_n^i &= \frac{1}{\pi}\int_{-\pi}^{\pi} \Lambda_L^i(\phi)\sin(n\phi)d\phi. \end{aligned} \tag{18}$$

a_n^i and b_n^i are Fourier coefficients for ith symbol as the overall LLF is the summation of the LLF of N individual symbols. This allows us to calculate the Fourier coefficients for every symbol separately.

In practice, we find that the higher harmonics are negligible in amplitude. For the purpose of maximization the constant a_0^i can also be ignored. Thus we approximate the LLF by ignoring the constant and higher harmonic terms.

1) BPSK: For BPSK, the Fourier series has only cosine items because its LLF (12) is periodic and even. Moreover, the

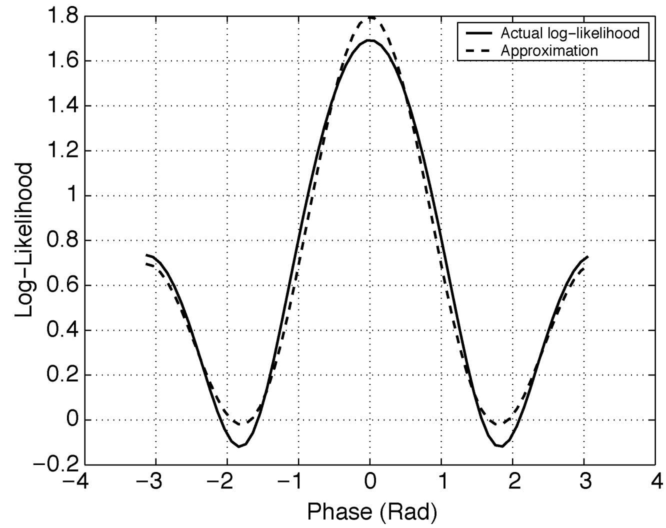


Fig. 1. Approximation of the LLF function using two harmonics Fourier series.

harmonics above the second are very small, so that the LLF is approximated as

$$\Lambda_{L,\mathrm{BPSK}}(\phi) \approx \sum_{i=1}^{N}\left[a_0^i + a_1^i\cos(\phi-\phi_i) + a_2^i\cos(2\phi-2\phi_i)\right] \tag{19}$$

where a_1^i and a_2^i are Fourier coefficients for the first and second harmonics of ith received symbol. This gives a reasonable approximation to the actual LLF, as shown in Fig. 1 for a typical case with $\phi_i = 0$.

Ignore the constant term a_0^i, then we express the complete LLF as

$$\begin{aligned} \tilde{\Lambda}_{L,\mathrm{BPSK}}(\phi) &= \sum_{i=1}^{N}\left[a_1^i\cos(\phi-\phi_i) + a_2^i\cos(2\phi-2\phi_i)\right] \\ &= w_1\cos(\phi-\theta_1) + w_2\cos(2\phi-\theta_2) \end{aligned} \tag{20}$$

where w_1 and w_2, the final magnitude of two harmonics, and θ_1 and θ_2, the final arguments, are obtained by phasor addition over the N components, such that

$$\begin{aligned} w_1\cos(\theta_1) &= \sum_{i=1}^{N} a_1^i\cos(\phi_i) \\ w_1\sin(\theta_1) &= \sum_{i=1}^{N} a_1^i\sin(\phi_i) \\ w_2\cos(\theta_2) &= \sum_{i=1}^{N} a_2^i\cos(2\phi_i) \\ w_2\sin(\theta_2) &= \sum_{i=1}^{N} a_2^i\sin(2\phi_i). \end{aligned} \tag{21}$$

We then find the ML estimate by setting the derivative of the LLF (20) equal to zero according to (16), giving

$$\frac{d\tilde{\Lambda}_{L,\mathrm{BPSK}}}{d\phi} = w_1\sin(\phi-\theta_1) + 2w_2\sin(2\phi-\theta_2) = 0. \tag{22}$$

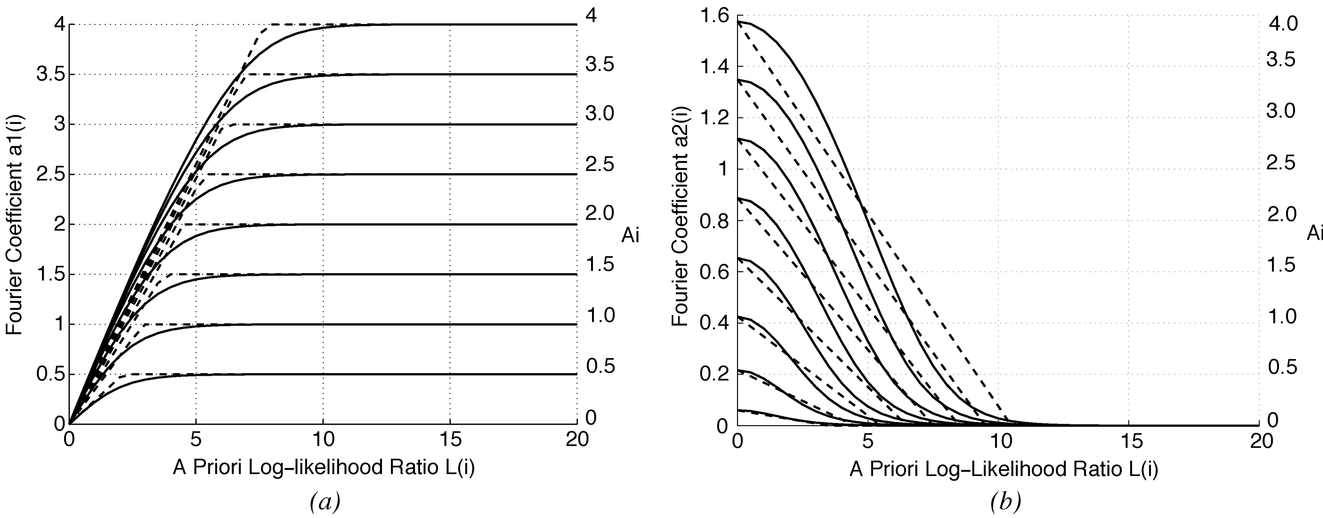


Fig. 2. Lookup table for Fourier coefficients a_1^i, a_2^i for BPSK. (a) First harmonic. (b) Second harmonic.

Normally, as ϕ approaches the desired phase estimate $\hat{\phi}_{ML}$, $\phi-\theta_1$ and $2\phi-\theta_2$ are very small and, hence, (22) can be approximated using $\sin x \simeq x$

$$\frac{d\tilde{\Lambda}_{L,\mathrm{BPSK}}}{d\phi} \simeq w_1(\phi-\theta_1)+2w_2(2\phi-\theta_2)=0. \tag{23}$$

The ML estimate $\hat{\phi}_{\mathrm{ML}}$ can then be easily attained by

$$\hat{\phi}_{\mathrm{ML}} = \frac{w_1\theta_1+2w_2\theta_2}{w_1+4w_2}. \tag{24}$$

This maximization is particularly simple if one of the harmonics is very large compared to the other.

However, the approximation only partially resolves the computation complexity problem. Since the Fourier coefficients, $\{a_1^i, a_2^i\}$, vary for each symbol with the integration operation inherent in the calculation, there is still a heavy computation burden. Our solution is to construct lookup tables for the values of $\{a_1^i, a_2^i\}$ as functions of A_i and $L(i)$. We found that the first harmonic coefficient a_1^i is odd symmetrical to $L(i)$, while a_2^i is even symmetrical. This allows us to only consider positive $L(i)$. Fortunately, the lookup tables are very simple as illustrated by the solid lines in Fig. 2. For further simplification, without significant loss of accuracy, they are replaced by segments (dotted lines in Fig. 2). Thus, only the gradients and crossing points of the linear sections need be stored. In addition, these gradients and crossing points can be obtained from linear equations depending on A_i and $L(i)$. In consequence, the lookup table only needs to store 4 data to establish the linear equations.

In this way, $\Lambda_{L,\mathrm{BPSK}}^i(\phi)$ can be worked out simply by linear and lookup operations. The complete LLF is the phasor summation over one block. As a result, all nonlinear calculation is avoided and the computation required is greatly reduced. The phase estimate is computed directly from the LLF without acquisition procedure. Therefore, no excessive complexity or delay is introduced to the system.

2) QPSK: In a QPSK system, the situation is more complex than in BPSK because two bits are transmitted per symbol. As expected, there are more harmonics, including $\{a_1^i, b_1^i, b_2^i, a_4^i\}$, in the Fourier series which cannot be ignored. Hence, the Fourier series can be written as

$$\tilde{\Lambda}_{L,\mathrm{QPSK}}(\phi) = \sum_{i=1}^{N}\left[a_1^i\cos(\phi-\phi_i)+b_1^i\sin(\phi-\phi_i)\right.$$
$$\left.+b_2^i\sin(2\phi-2\phi_i)+a_4^i\cos(4\phi-4\phi_i)\right] \tag{25}$$

where $\{a_4^i\}$ is significant only when there is no *a priori* information available. In this case, both $L_u(i)$ and $L_c(i)$ are equal to 0, and the Fourier series reduces to

$$\tilde{\Lambda}_{L,\mathrm{QPSK}}(\phi) = \sum_{i=1}^{N}\left[a_4^i\cos(4\phi-4\phi_i)\right]. \tag{26}$$

Note that this is equivalent to the NDA method and $\{a_4^i\}$ only depends on A_i, i.e., the amplitude of the ith received signal by two linear functions. These two linear functions are stored in a lookup table, from which a_4^i can be easily computed given the received signal.

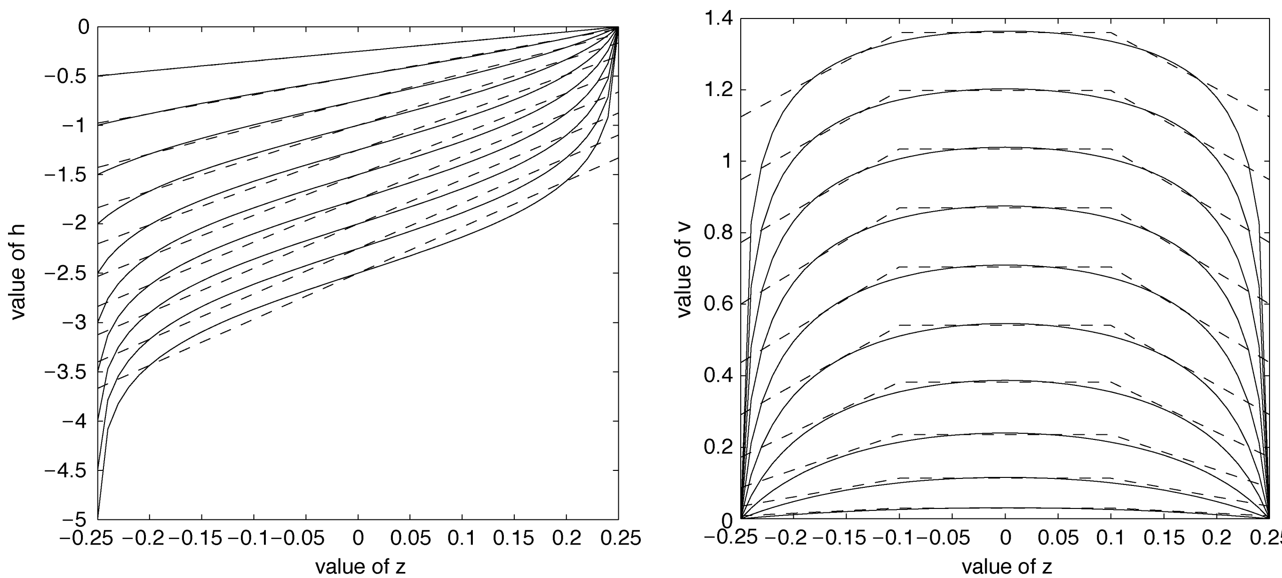
The other three harmonics $\{a_1^i, b_1^i, b_2^i\}$ are determined by $L_u(i)$, $L_c(i)$, as well as A_i. Although more effort is inevitable to discover the relationships between them, the lookup tables turn out to be linear and simple and only small amount of data needs to be stored as desired.

Define parameter z as

$$z=\frac{1}{4}\tan h\left(\frac{L_u(i)}{2}\right)\tan h\left(\frac{L_c(i)}{2}\right). \tag{27}$$

We obtain parameter h for the first harmonic coefficients and v for the second harmonic in relation to z as plotted in Fig. 3. The parallel curves result from different values of A_i. The first harmonic Fourier coefficients $\{a_1^i, b_1^i\}$ can be easily obtained from h by simple operations like parallel shift and π rotation (if needed) while the coefficient $\{b_2^i\}$ is similarly obtained from v. We will not explore the details due to the limitation of space.

Employing these lookup tables, nonlinear computation is totally replaced by linear and lookup operations. The ML estimate



(a) *(b)*

Fig. 3. Lookup table for Fourier coefficients a_1^i, b_1^i, b_2^i for QPSK. (a) First harmonic. (b) Second harmonic.

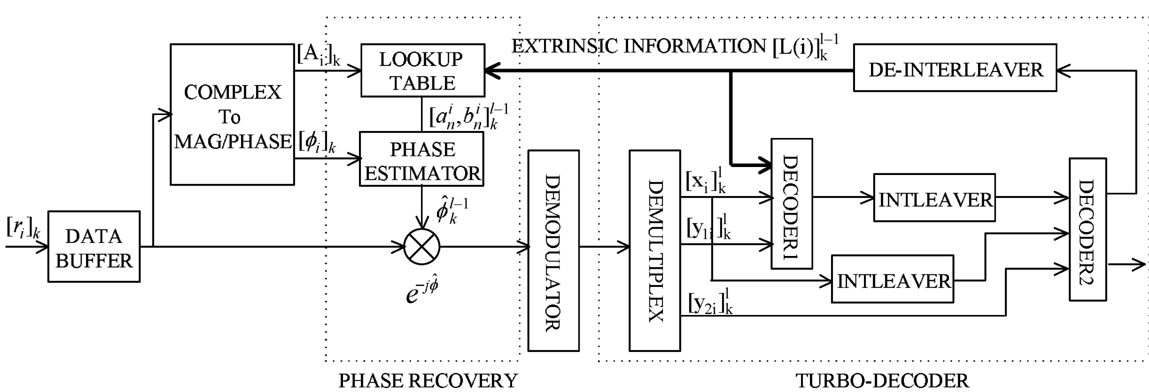


Fig. 4. Structure of iterative joint carrier phase recovery and turbo decoding.

is then obtainable directly from the complete LLF, which has been simplified as

$$\tilde{\Lambda}_{L,\mathrm{QPSK}}(\phi) = w_4 \cos(4\phi - \theta_4) \tag{28}$$

for the first iteration, when no *a priori* information is available, or

$$\tilde{\Lambda}_{L,\mathrm{QPSK}}(\phi) = w_1 \cos(\phi - \theta_1) + w_2 \cos(2\phi - \theta_2) \tag{29}$$

in other iterations, where w_1, w_2, w_4, and $\theta_1, \theta_2, \theta_4$ are obtained from the phasor addition over one block.

C. Diagram

Summarizing the above description, we illustrate this approach in Fig. 4. The APPA phase estimation is a block based iterative synchroniser combined with the turbo decoding. These two procedures operate as follows: the kth received block, expressed as $[r_i]_k$, is fed to the phase estimator and the phase corrector (phase rotator $e^{-j\phi}$) simultaneously, and the corrector corrects the input sequence using the phase estimate produced by the estimator in last iteration $\hat{\phi}_k^{l-1}$ (l denotes current iteration number), which is initialized as 0 in the first iteration of every block. The corrected sequence $[A_i e^{j(\phi_i - \hat{\phi}_k^{l-1})}]_k$ is sent to the turbo decoder for the lth decoding iteration. Meanwhile, the phase estimator calculates a phase estimate for the current (lth) estimation iteration using $[A_i]_k$ and $[L(i)]_k^{l-1}$ (note $[L(i)]_k^{l-1}$ is also initialized as 0 in the first iteration, equivalent to the NDA method). The received block will be kept unchanged for K iterations using a data buffer, but the *a priori* information $[L(i)]_k^{l-1}$, the phase estimate $\hat{\phi}_k^{l-1}$ as well as the data fed into the decoder are updated every iteration. Operated iteratively, the decoding improves the phase estimation by providing more reliable $[L(i)]_k^{l-1}$, while the phase estimation improves the decoding by generating more accurate estimates, until the synchronization/decoding converges after enough iterations (i.e., the number of iterations that the receiver needs for the synchronization and decoding to converge, in our cases, six iterations).

IV. Simulation Results

The APPA technique has been tested using a classical rate half-turbo-code which consists of a parallel concatenation of two identical 16-state RSC constituent codes with generators $G_1 = (023)_8$, $G_2 = (037)_8$ linked by a length N-bit S-random interleaver, with regular, alternate puncturing of the parity streams. Interleaver sizes of $N = 256$, $N = 512$, and $N = 1024$, with S-parameters of $s = 11$, $s = 15$, and $s = 19$, respectively are considered. The turbo-decoder is based on the Log-MAP algorithm [11]. The channel is modeled as AWGN. Initially, we assume that the carrier phase error is unknown but constant over one block. It is worth mentioning here that the algorithm has been tested and found to be able to tolerate a frequency offset less than $10^{-4}/T$ (T is the symbol period). This value could be increased by partitioning the code block into shorter estimation windows (sub-blocks), over which our estimation algorithm is applied. This can also be used to combat slow fading, where the phase error is varying within a block. It is reasonable to assume the phase error is constant over shorter estimation windows on slow fading channels. However, this method is not robust for large frequency offsets and fast fading channels.

Conventional methods, based on the squaring loop [13] and the fourth-power law method [15] were also developed to compare with the APPA method. The received signal is squared (or taken to the fourth power) to remove the effect of the data modulation. The average phase of the result is divided by two (or four) to remove the effect of the squaring (or fourth-power). This is done over one block to estimate the phase error. The symbol is then corrected using this phase estimate before being fed into the decoder.

The performance of this technique is evaluated individually for BPSK and QPSK systems in terms of mean phase estimate, MSEE, overall BER versus phase error, and BER versus energy per bit to noise spectral density ratio E_b/N_0. The BER performance is evaluated in comparison with the traditional NDA method and the performance of the optimally synchronised system. We also measure the accuracy of the estimator by comparing the mean-square estimation error (MSEE) with the Cramer-Rao bound (CRB) and the result of recent similar work [9].

A. BPSK

The BER performance for BPSK with phase errors in the region of $[-180°, 180°]$ is shown in Fig. 5. The three curves in this figure correspond to three different block sizes: $N = 256$, $N = 512$, and $N = 1024$. The two flat regions on each curve are obvious: the lower one demonstrates that the system provide reliable performance with any phase error in this region; we call it the normal operating region. It is shown that this technique ensures that the decoding is robust against a remarkably wide range of phase errors. The largest phase error from which the system can recover for size $N = 1024$ is $\pm 82°$. The system fails quickly, however, when the phase error increases beyond this operating region. The upper flat region appears when the phase error $|\phi| > 90°$, where we note that the BER approaches

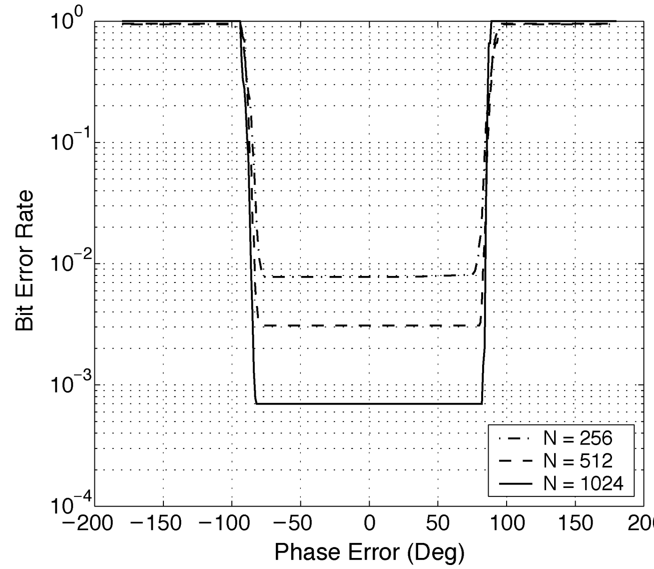


Fig. 5. BER performance against phase errors for BPSK system, $E_b/N_0 = 1.5$ dB, 4 iterations.

1. This can be interpreted by considering that for BPSK modulation, the ML phase estimator has two ambiguity estimates 180° apart. With a phase error $|\phi| > 90°$, the opposite constellation position is chosen in error. In BPSK, this inverts the transmitted data bits. As a result, the final decoding decision may not converge, or may converge to a code word in which all data bits inverted, after several iterations of the joint synchronization and decoding. Clearly, for a longer block size, the normal operating area is wider, as well as giving a better decoding performance (lower BER level). However, a longer block size will also introduce more delay. Hence, there is a tradeoff between extra delay and better performance. In the following, all the other simulation results are obtained using a turbo code with block size $N = 1024$.

Some researchers, such as Wiberg [3], have investigated the use of rotationally invariant turbo codes to make the decoding robust against phase uncertainty. We have found that turbo-coded BPSK, using the above particular turbo codes, does not have the rotational invariance property. In fact, no codeword is obtained by a rotation of another codeword. This allows us to employ a different approach to deal with phase errors larger than 90°; we use two identical joint phase estimation and decoding units operating with phase references π apart. The decoding with the higher metric (mean square LLR over a block) is selected as the final decision. Similar techniques were also developed for the QPSK system using four identical joint phase estimation and decoding units $\pi/2$ apart. The resulting system can then provide reliable phase estimation and robust decoding against nearly any phase errors, at the price of implementation complexity. The results have been reported in [16] and [17] respectively, but space does not allow their exploration here.

Fig. 6 depicts the BER-E_b/N_0 performance of the APPA technique in comparison with the squaring loop and the optimally synchronized system. BER curves obtained from four iterations at three phase error values 15°, 40°, 85° are plotted. Note that for phase error 85°, the BER is very sensitive, and

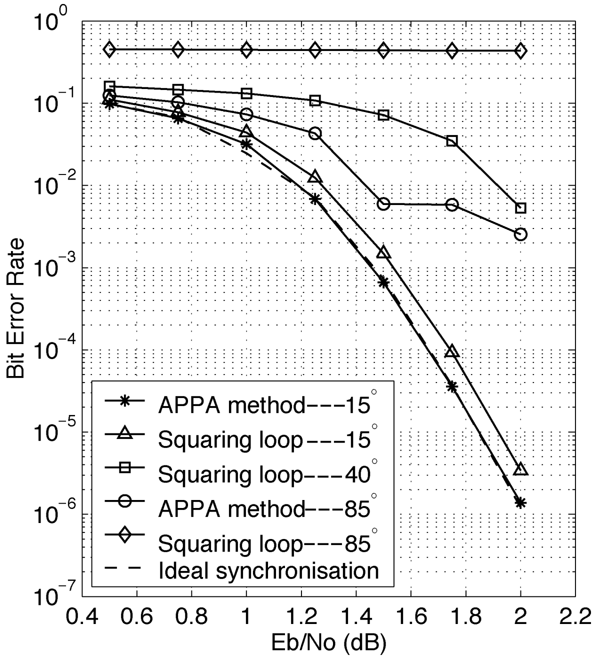


Fig. 6. BER-E_b/N_0 performance for BPSK system in comparison with the squaring loop at phase offsets 15°, 40°, 85°, four iterations.

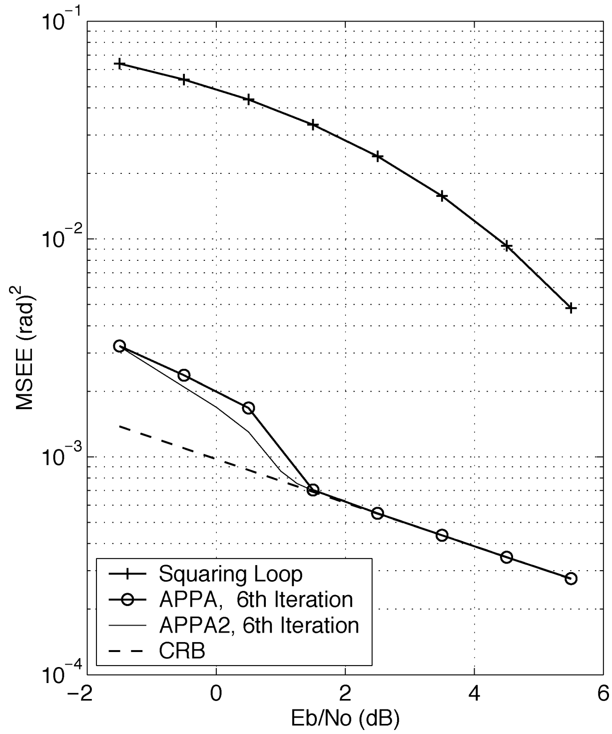


Fig. 7. Mean-square estimation error (MSEE) for turbo-coded BPSK with a phase error of 20°.

therefore, random variations appear on the corresponding trace, which is less smooth than the others. It is clear, however, that this does not affect our conclusions which follow. The BER performance of APPA is very close to the ideal when the phase error is less than 82°, and greatly outperforms the squaring loop. We also find that the improvement is larger at larger phase error. This can be interpreted with reference to the flat region in Fig. 5, which shows that the BER performance of the APPA method is constant for all phase errors less than 82°. This, however, is not the case for the squaring loop, the BER for which increases with larger phase errors. This is because of the increased probability that the phase estimator selects the wrong point (180° relative to the desired phase estimate) due to the two fold phase ambiguity in the presence of the increased noise power resulting from the squaring operation. Thus, the improvement increases at larger phase errors.

This method is also superior to a most recently proposed joint decoding and carrier phase recovery algorithm for turbo-coded BPSK system [8]. Unlike our technique, the steady-state BER performance of this algorithm did not match that with perfect phase estimation, but has a 0.1-dB SNR loss at BER 10^{-3}.

Cramer–Rao bounds (CRBs) [18] give the theoretical lower limits to the variance of any unbiased parameter estimator. It is used to measure the accuracy of the estimation. We compared the MSEE, i.e., $E\{(\hat{\phi}-\phi)^2\}$, of the APPA and the squaring loop phase estimators with the CRB for any unbiased estimator [19]

$$E\{(\hat{\phi}-\phi)^2\} \geq \frac{1}{N\frac{E_b}{N_0}}. \quad (30)$$

Fig. 7 plots the curves. The APPA method achieves the bound at merely $E_b/N_0 \geq 1.5$ dB after six iterations, showing that the technique provides a performance as reliable as the ideal when the E_b/N_0 is higher than 1.5 dB. From the comparison in this figure, the APPA method also demonstrates a significant improvement over the squaring loop method in terms of the accuracy of estimation. The curve labeled "APPA2" is obtained by a modified APPA method which makes use of the sum of the extrinsic information obtained from both decoders. Clearly, extra extrinsic information explicitly reduced the MSEE for $E_b/N_0 < 1.5$ dB. However, it does not significantly affect the BER and mean of the phase estimate. Therefore, because of practical considerations, it was not adopted here.

B. QPSK

The same length 1024-bit turbo codes are used for QPSK system. Bits are mapped a pair at a time onto the QPSK constellation using Gray mapping. Since the rate half-turbo-code is obtained by regular, alternate puncturing the parity streams, one of each pair of bits is a data bit, the other is a parity bit. Thus, extrinsic information for parity bits is also required in QPSK. We calculate it directly within the MAP decoder, in the same way as that of the data bits.

QPSK modulation provides higher spectrum efficiency than BPSK, however, more points in the constellation also reduce the decision region, hence, QPSK is more sensitive to phase errors. Therefore, as we expected, the BER performance versus phase errors appears the same shape as in the BPSK system with a much narrower normal operating region, between $|\phi| \leq 38°$, as illustrated in Fig. 8. Note that the BER level is half when $|\phi| > 45°$ rather than 1 as for BPSK (refer to Fig. 5). The explanation is similar to that for BPSK; considering that a rate half-turbo-code was used, only one bit of the QPSK symbol is a data bit,

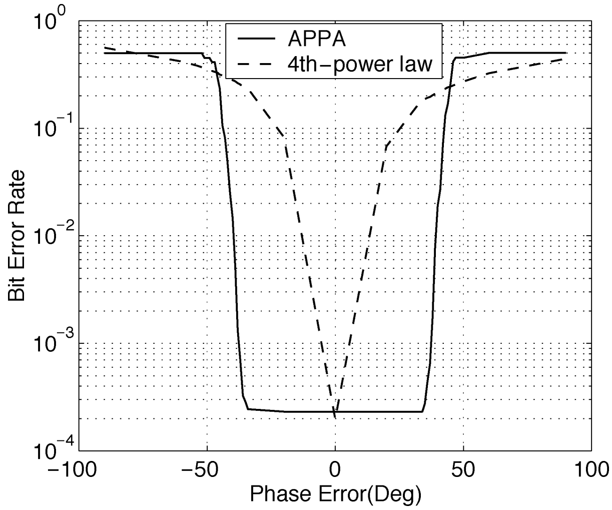


Fig. 8. BER against phase offsets for QPSK system in comparison with fourth-power law method at $E_b/N_0 = 1.5$ dB, 8 iterations.

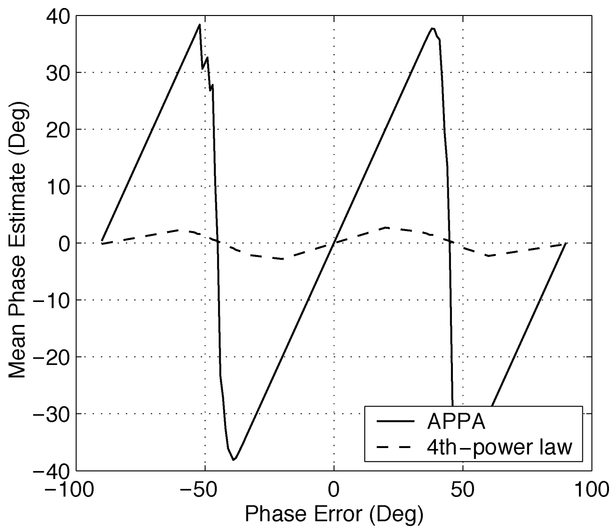


Fig. 9. Mean phase estimate against phase offsets for QPSK system in comparison with fourth-power law method at $E_b/N_0 = 1.5$ dB, 8 iterations.

and thus, only half of the phase rotations larger than 45° invert the data bit, and hence, the BER is approximately half.

For the purpose of comparison, in the same figure, we also illustrate the BER performance of the fourth-power law carrier recovery method, which is obviously much worse than the APPA method. This result is demonstrated by the mean phase estimate curves of these two methods shown in Fig. 9. The curve of the APPA method appears as a straight line with gradient very close to 1 between $|\phi| \leq 38°$, which means that the phase estimator produces an accurate estimate in this range. However, out of this range the phase estimation is completely inaccurate which causes the failure of the system. In contrast, the fourth-power law method has poor estimation even for small phase errors.

Fig. 10 depicts the accuracy of the APPA phase estimator for QPSK by comparing with the CRB and similar work from the literature. As mentioned in the introduction, a recent paper of Lottici *et al.* [9] proposes a joint phase estimation and turbo decoding scheme, called ISDD, where a rate half-turbo-code with generators $G_1 = (31)_8$ and $G_2 = (33)_8$, via a pseudo-random interleaver with length of 1500 is used. To compare with his work, we convert the MSEE to root-mean-square estimation error (RMSEE) in degrees. The CRB is then written as [18]

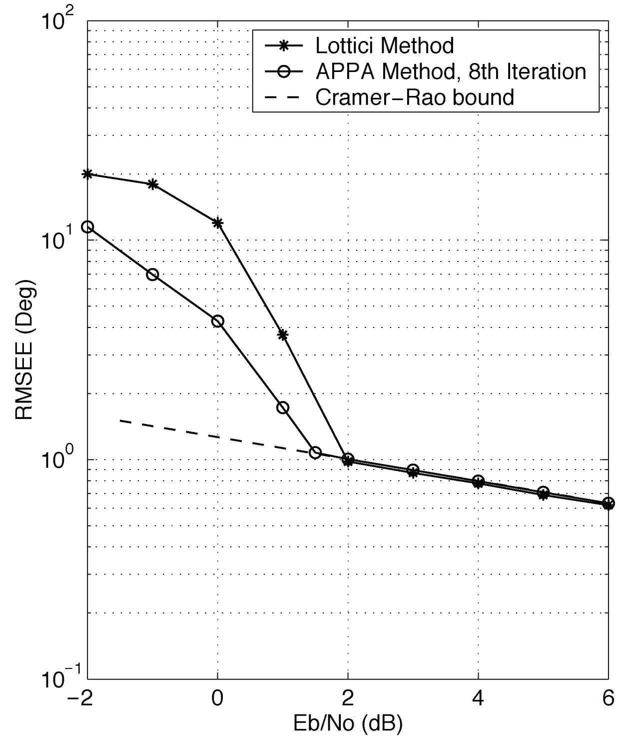


Fig. 10. Root Mean Square Estimation Error (RMSEE) for turbo-coded QPSK system.

$$\text{RMSEE} \geq \frac{180°}{\pi\sqrt{2N\frac{E_b}{N_0}}}. \tag{31}$$

Both systems approach the CRB at very low SNRs, but the APPA has 0.5 dB improvement over the ISDD. Both methods compute the phase estimate $\hat{\phi}$ directly from the LLF without an acquisition procedure. In ISDD, this is implemented through approximating the log and exp functions into linear functions. This allows an approximation to the phase of the fundamental of the LLF to be calculated, but does not accurately model the LLF. In addition, the technique cannot operate with no *a priori* information, and hence, cannot operate in the first iteration, for which a separate NDA estimator is required. In the APPA, Fourier expansion is used, which takes into account all the significant coefficients using lookup tables to avoid heavy computation while causing negligible degradation to the performance. Although the space prohibits the inclusion of the BER performance here, it is worth mentioning that the simulation results showed that the overall BER performance of the turbo-coded QPSK system with APPA phase recovery matches that with ideal synchronization, whereas the BER of the turbo-coded 4-QAM with ISDD phase recovery after ten iterations is about 0.2 dB away from the ideal at a BER of 10^{-5}.

V. CONCLUSION

In this paper, we present an alternative iterative carrier phase estimation concept suited to turbo-coded systems. It exploits the extrinsic or so-called "*a priori*" information generated at each iteration of the turbo decoder, and hence, we have named it *a priori* probability aided (APPA) phase estimation. The extrinsic information, which is usually used as the *a priori* information to improve the decoding, is utilized in the phase estimation algorithm through a joint log-likelihood function. The phase estimation and the turbo decoding operate jointly and iteratively. In this way, this technique allows a successive refinement of the carrier phase estimate until the joint decoder/synchroniser converges on the (hopefully) correct data and phase estimate. No excessive complexity is added to this system since the complexity of this method is significantly reduced by expanding the LLF as a Fourier series whose coefficients are precalculated and stored in lookup tables. Hence, the phase estimate can be computed directly from the LLF without an acquisition procedure.

It has been shown that this technique ensures robust decoding against a wide range of phase errors and reliable estimation in which the MSEE attains the CRB at very low SNR values (1.5 dB), meaning that the estimator is as accurate as possible, even at low SNRs. It is also demonstrated that the APPA phase estimation greatly outperforms the traditional NDA methods and achieves the performance of ideal synchronization without introducing the additional redundancy of a synchronization preamble.

The technique presented can be straightforwardly extended to other similar processes, such as symbol timing recovery, frequency estimation, multiuser detection, and channel estimation, etc. The application with higher order modulation and time varying channels will be subject of future research.

REFERENCES

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo codes," in *IEEE Int. Conf. Commun.*, vol. 2, Geneva, Switzerland, May, 23–26 1993, pp. 1064–1070.

[2] A. Risley, B. Belzer, and Y. Zhu, "Turbo trellis coded modulation on partially coherent fading channels," in *Proc. 2000 IEEE Int. Symp. Information Theory (ISIT2000)*, Sorrento, Italy, June 25–30, 2000, p. 222.

[3] N. Wiberg, "Simultaneous decoding and phase synchronization using iterative "turbo" decoding," in *IEEE Int. Symp. Information Theory*, Ulm, Germany, June 29-July 4, 1997, p. 11.

[4] B. Mielczarek and A. Svensson, "Phase offset estimation using enhanced turbo decoders," in *IEEE Conf. Commun.*, vol. 3, New York, Apr. 28–May 2, 2002, pp. 1536–1540.

[5] C. Komninakis and R. D. Wesel, "Joint iterative channel estimation and decoding in flat correlated rayleigh fading," *IEEE J. Sel. Areas Commun.*, vol. 19, pp. 1706–1717, Sept. 2001.

[6] C. Morlet, I. Buret, and M.-L. Boucheret, "A carrier phase estimator for multi-media satellite payloads suited to RSC coding schemes," in *IEEE Int. Conf. Commun.*, vol. 1, New Orleans, LA, 2000, pp. 455–459.

[7] C. Langlais and M. Helard, "Phase carrier for turbo codes over a satellite link with the help of tentative decisions," in *Int. Symp. Turbo Codes Related Topics*, vol. 5, 2000, pp. 439–442.

[8] W. Oh and K. Cheun, "Joint decoding and carrier phase recovery algorithm for turbo codes," *IEEE Commun. Lett.*, vol. 5, pp. 375–377, Sept. 2001.

[9] V. Lottici and M. Luise, "Iterative carrier phase synchronization for coherent detection of turbo-coded modulation," in *Euro. Wireless 2002*, Florence, Italy, Feb. 25-28, 2002, pp. 840–845.

[10] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform Theory*, vol. IT-20, pp. 284–287, Mar. 1974.

[11] P. Robertson and P. Hoeher, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding," *Eur. Trans. Telecom*, vol. 8, no. 2, pp. 119–125, 1997.

[12] W. E. Ryan. (1998) A Turbo Code Tutorial. [Online]. Available: http://citeseer.nj.nec.com/ryan97turbo.html

[13] J. G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1995.

[14] G. Colavolpe, G. Ferrari, and R. Raheli, "Extrinsic information in iterative decoding: A unified view," *IEEE Trans. Commun.*, vol. 49, pp. 2088–2094, Dec. 2001.

[15] L. E. Franks, *Digital Communications: Satellite/Earth Station Engineering*, K. Feher ed: Prentice-Hall, 1983, ch. Synchronization subsystems: Analysis and design, pp. 294–317.

[16] L. Zhang and A. Burr, "A new method of carrier phase recovery for bpsk system using turbo codes over awgn channel," in *IEEE Personal, Indoor, Mobile Radio Commun. Conf. (PIMRC)*, vol. 1, San Diego, CA, Sept. 30–Oct. 3, 2001, pp. A-179–A-183.

[17] ——, "A novel carrier phase recovery method for turbo-coded qpsk system," in *Proc. Eur. Wireless 2002*, Feb. 2002, pp. 817–821.

[18] U. Mengali and A. N. D'Andrea, *Synchronization Techniques for Digital Receivers*. New York: Plenum, 1997.

[19] D. C. Rife and R. R. Boorstyn, "Single-tone parameter estimation from discrete-time observations," *IEEE Trans. Inform. Thoery*, vol. IT-20, pp. 591–598, Sept., 1974.

Li Zhang received the B.Sc. degree in electronic engineering and the M.Eng. degree in signal and information processing from the Department of Electronics, Beijing University of Aeronautics and Astronautics, Beijing, China, in 1995 and 1998, respectively. She received the Ph.D. degree from the Communications Research Group, Department of Electronics, University of York, U.K., in 2003.

From 1999 to 2000, she worked as a Research Scientist/Software Engineer at China Software Design Centre, Agilent Technologies Ltd., Beijing, China. She is currently a Lecturer in communications with the University of Leeds, Leeds, U.K. Her research interests lie in the areas of wireless communications and signal processing, including synchronization, coding and modulation, turbo-codes, MIMO systems, and space-time coding.

Alister G. Burr was born in London, U.K., in 1957. He received the B.Sc. degree in electronic engineering from the University of Southampton, Southampton, U.K., in 1979 and the Ph.D. degree from the University of Bristol, Bristol, in 1984.

Between 1975 and 1985, he worked at Thorn-EMI Central Research Laboratories, London, U.K. In 1985, he joined the Department of Electronics, University of York, York, U.K., where he has been Professor of communications since 2000. He has also held a visiting professorship at the Vienna University of Technology, Vienna, Austria. His research interests are in wireless communication systems, especially modulation and coding and including turbo-codes and turbo-processing techniques, MIMO systems and space-time coding.

Dr. Burr was awarded a Senior Research Fellowship by the U.K. Royal Society in 1999. In 2002, he received the J. Langham Thompson Premium from the Institution of Electrical Engineers. He is currently Chair, Working Group 1, of the European COST 273 program "Toward Broadband Mobile Networks."