EXTENDED CONSTRAINED VITERBI ALGORITHM FOR AIS SIGNALS RECEIVED BY SATELLITE

Raoul Prévost[1,2]*, Martial Coulon*[1]*, David Bonacci*[2]*, Julia LeMaitre*[3]*,*
Jean-Pierre Millerioux[3] *and Jean-Yves Tourneret*[1]

[1] University of Toulouse, INP-ENSEEIHT/IRIT, 2 rue Charles Camichel, BP 7122, 31071 Toulouse cedex 7, France
[2] TéSA, 14-16 Port Saint-Étienne, 31000 Toulouse, France [3] CNES, 18 Avenue Edouard Belin, 31400 Toulouse, France
{raoul.prevost, david.bonacci}@tesa.prd.fr, {martial.coulon, jean-yves.tourneret}@enseeiht.fr,
{julia.lemaitre, jean-pierre.millerioux}@cnes.fr

ABSTRACT

This paper addresses the problem of error correction of AIS messages by using the a priori knowledge of some information in the messages. Indeed, the AIS recommendation sets a unique value or a range of values for certain fields in the messages. Moreover, the physics can limit the range of fields, such as the speed of the vessel or its position (given the position of the receiver). The repetition of the messages gives also some information. Indeed, the evolution of the ship position is limited between messages and the ship ID is known. The constrained demodulation algorithm presented in this article is an evolution of the constrained Viterbi algorithm (C-VA). It is based on a modified Viterbi algorithm that allows the constraints to be considered in order to correct transmission errors by using some new registers in the state variables. The constraints can be either a single value or a range of values for the message fields. Simulation results illustrate the algorithm performance in terms of bit error rate and packet error rate. The performance of the proposed algorithm is 2 dB better than that obtained with the receiver without constraints.

Index Terms— AIS, Satellite, constraints, bit-stuffing, CRC, Viterbi decoding, C-VA.

1. INTRODUCTION

This paper addresses the problem of demodulating messages received by a satellite transmitted by the automatic identification system (AIS) [1]. AIS is a self-organized TDMA access system, whose objective is to avoid collision of large vessels. The AIS system allows the ships to share information about their location and speed with frames composed of 168 information bits concatenated with a 16-bit cyclic redundancy check (CRC). In order to avoid the presence of end frame marker inside the frame data, stuffing bits (corresponding to bits 0) are inserted in the frame after each sequence of five bits 1. After the bit stuffing procedure, start and end markers are inserted at the beginning and the end of the frame. The final binary sequence is encoded in non-return-to-zero inverted (NRZI), and modulated using the Gaussian minimum shift-keying (GMSK). Basically, the AIS is not designed for a satellite reception. However, the demodulation of AIS signals received by a satellite would be useful for the global supervision of the maritime traffic. To that purpose, new correction methods have to be developed in order to obtain acceptable packet error rates at low E_b/N_0.

An efficient demodulation algorithm has been recently proposed in [2] in the context of a single-user scenario with an interesting solution involving a modified Viterbi algorithm. In [2] the cyclic redundancy check (CRC) contained in the AIS messages is used as a source of redundancy to correct transmission errors in presence of bit stuffing. Other solutions were proposed previously (see [2], [3], [4], [5] and references therein) to correct errors by using the CRC as redundancy, and not only as an error detection tool, as it was primarily conceived. However, these methods cannot be used in presence of bit stuffing. An interference mitigation strategy has also been proposed in [6] for the multi-user case.

The receiver proposed in [2] showed important enhancements of the performances with respect to those obtained with the conventional approach, based on GMSK coherent demodulation. However, improvements of these receivers can still be investigated. In particular, the novelty of this paper consists of using the knowledge of some information contained in the messages and specific characteristics of the AIS standard. Indeed, the AIS recommendation sets a unique value or a range of values for certain fields in the messages. Moreover, the physics can limit the range of fields such as the speed of the vessel or its position (given the position of the receiver). The repetition of the messages gives also some information. Indeed, the evolution of the ship position is limited between messages, and the ship ID is known. Taking into account this recommendation in the design of the demodulation algorithm allows supplementary constraints to be defined which can be used to correct transmission errors. The constrained Viterbi

The authors would like to thank the DGA and the CNES for funding.

algorithm (C-VA) [7] allows the demodulator to take into account some constraints. However, the C-VA is restricted to the cases where the fields constraints can only take one single value, and, moreover, it is not suitable for the bit stuffing procedure. The proposed receiver uses a constrained demodulation algorithm based on the modified Viterbi algorithm proposed in [2] and is designed to take into account the bit stuffing operation inherent to the AIS system. In order to deal with the constraints, the algorithm presented in this paper uses registers that are added to the Viterbi algorithm states. These registers allow one to store the received bits along the paths in the trellis, and to use the stored values to check the validity of the corresponding paths. The copy of the received bits in the registers and the checking of the register values are made according to a list of instructions that associate the number of received information bit with the actions to be performed.

The paper is organized as follows. The AIS transmission scheme is described in Section 2. The main principles of the demodulation algorithm are reported in section 3. Section 4 presents the constraints considered for the AIS system. Section 5 describes the instructions performed on the trellis transitions to improve the performance of the basic algorithm. Section 6 details the list of instructions associated with the constraints. Some simulation results obtained from a realistic AIS simulator (developed by the CNES of Toulouse, France) are presented in Section 7. Conclusions are finally reported in Section 8.

2. TRANSMISSION SCHEME

The AIS transmission scheme is illustrated in Fig. 1: 168 information bits are transmitted. A CRC is then computed and concatenated to these bits. The bit stuffing procedure is applied on the resulting sequence. The final binary sequence is encoded in non-return-to-zero inverted (NRZI), and modulated using the Gaussian minimum shift-keying (GMSK) modulation.

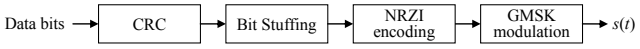

Fig. 1. Transmitter model.

2.1. CRC Computation and Bit stuffing

It is well known that the CRC is defined as the remainder of the division (modulo 2) of the polynomial formed by the data and a standardized generator polynomial, whose degree equals the length of the CRC plus one. Some zeros are generally added before the remainder to obtain a fixed-length CRC. At the receiver side, the CRC is computed from the received data, and compared with the CRC computed with the original data, which is added to the information bits before transmission. One or more errors are detected in the transmitted data if both CRCs are not identical. An important property of the CRC is that its computation can be performed iteratively by initializing the CRC to a standard value and by applying the operations to each data bit. This property is crucial for the strategy proposed in this paper, since the trellis is developed thanks to this iterative CRC computation.

Moreover, when the CRC is transmitted after the information bits, the receiver can compute a joint CRC on the sequence composed of the information bits and the CRC. Thus, instead of comparing two CRCs computed separately, there is no error when the joint CRC is zero such that

$$\mathrm{CRC}([\mathrm{Data}, \mathrm{CRC}(\mathrm{Data})]) = 0. \tag{1}$$

In addition to the CRC, some non-informative bits called stuffing bits can be inserted into the information message. The insertion of stuffing bits presents two main advantages: i) the generated additional transitions allow the receiver to resynchronize its clock; ii) it avoids some specific bit sequences, such as begin or end flags (composed of two bits 0 on each side of six consecutive bits 1 in the AIS system). Since only bits 0 are inserted in that case, this particular bit stuffing is called zero-bit insertion. In this paper, it is assumed for notation convenience, that the stuffing bit is always a bit 0, as specified for AIS.

2.2. GMSK modulation

The bit sequence obtained after the bit stuffing procedure is encoded using the NRZI coding. The resulting sequence is modulated with the GMSK modulation. In the GMSK modulation, the transmitted signal $s(t)$ is a constant-modulus signal, which is expressed as

$$s(t) = \exp\left(-j2\pi h \sum_{k=-\infty}^{n} b_k q(t - kT)\right) \tag{2}$$

for $nT \leqslant t \leqslant (n+1)T$, where T is the symbol period, $(b_k)_k$ is the bit sequence, h is the modulation index, and $q(t)$ is the GMSK waveform [1].

3. PRINCIPLES OF THE BASIC ALGORITHM

This section presents briefly the proposed detection algorithm (see [2] for the complete presentation) from which the extended constrained Viterbi algorithm is developed.

3.1. General principle

Consider a frequency-flat transmission channel, whose transmission delay, Doppler and phase shifts are known by the receiver. The received signal can be expressed without loss of generality as

$$r(t) = s(t) + n(t) \tag{3}$$

where $s(t)$ is the signal generated at the output of the encoding-plus-modulation block, as depicted in Fig. 1, and $n(t)$ is a white additive Gaussian noise, independent of the transmitted data. The received signal (3) is first passed through a

matched filter and sampled with one sample per symbol. Let r_k denotes the resulting sample obtained for the kth symbol period. The standard Viterbi algorithm minimizes the square Euclidean distance between the received samples and the estimated symbols defined as

$$d^2 = \sum_{k=1}^{K} |r_k - m_k|^2 \tag{4}$$

where K is the number of received symbols, and m_k is the sample of the kth estimated symbol after matched-filtering. The proposed algorithm is based on a constrained maximum likelihood estimator minimizing the square Euclidean distance defined in (4) subjected to two constraints: C_1) the number of consecutive ones is upper bounded by a maximum value $\overline{P}$ specified by the standard, C_2) the CRC satisfies (1). In order to satisfy these constraints, the proposed receiver is based on a trellis composed of extended states formed by a CRC state and a trellis coding (TC) state.

The trellis is designed so that all paths ending with a final state give a message whose joint CRC is zero, according to (1) (note that the paths corresponding to a non zero CRC do not appear in the trellis). Moreover, the stuffing bits are taken into account by considering specific transitions in the extended trellis.

3.2. Trellis design

Since the CRC can be computed iteratively, it can be initialized depending on the AIS standard, and updated for every received bit. The CRC states are then defined as the intermediate CRC values. Two consecutive CRC states are linked if the second CRC can be obtained from the first one by updating the first CRC with one bit 0 or 1.

The algorithm proposed in [2] is based on a so-called extended trellis, where each state is composed of a CRC state and a TC state. These extended states are denoted $(A;\alpha)$ where A is the CRC state and α is the TC state.

In order to perform a Viterbi algorithm adapted to this extended trellis, the distance $\Gamma[k,(A;\alpha)]$ is defined as the distance between the received signal and the sequence of k symbols coming to the extended state $(A;\alpha)$ at time k, i.e.,

$$\Gamma[k,(A;\alpha)] = \sum_{i=1}^{k} |r_i - m_i^{k,(A;\alpha)}|^2 \tag{5}$$

where $m_1^{k,(A;\alpha)}, \ldots, m_k^{k,(A;\alpha)}$ denotes the symbol sequence reaching $(A;\alpha)$ at time k. Moreover, $\Gamma_{trans}[k,(A;\alpha),b]$ is the transition variable defined as the sum of $\Gamma[k,(A;\alpha)]$ and the squared distance between the received symbol at time $k+1$ and the symbol coming from the extended state $(A;\alpha)$ containing the bit b, denoted by $m_k^{k+1,(A;\alpha),b}$. More explicitly, one has

$$\Gamma_{trans}[k,(A;\alpha),b] = \Gamma[k,(A;\alpha)] + \Delta[k,(A;\alpha),b] \tag{6}$$

with

$$\Delta[k,(A;\alpha),b] = |r_k - \boldsymbol{m}_k^{k+1,(A;\alpha),b}|^2. \tag{7}$$

These transition variables of the form $\Gamma_{trans}[k,(A;\alpha),b]$ are used to choose the transition which leads to a given state, among the different possible transitions leading to this state, as detailed in [2].

3.3. Bit stuffing

In order to take bit stuffing into account, specific transitions are defined in the extended trellis. These transitions only occur when a stuffing bit is received, which requires the receiver to decide if a received bit is a stuffing bit or not. To that end, each extended state $(A;\alpha)$ is assigned a state variable $P[k,(A;\alpha)]$, which is defined as the number of consecutive bits 1 received before the state $(A;\alpha)$ at time k. The received bit at time k is declared as a stuffing bit when $P[k,(A;\alpha)]$ reaches a fixed maximum value $\overline{P}$ ($\overline{P} = 5$ for AIS): in that case, the only possible transition from $(A;\alpha)$ is the specific transition. After this transition, $P[k+1,(A;\beta)]$ takes the value 0 (recall that only stuffing bits equal to 0 are considered in this paper).

Finally, each extended state $(A;\alpha)$ is assigned a state variable $S[k,(A;\alpha)]$, defined as the number of stuffing bits received before reaching $(A;\alpha)$, which indicates the number of informative bits in the received frame. This variable, along with some other variables, allows one to determine the optimal path in the extended trellis, respecting the constraints C_1 and C_2, as detailed in [2].

4. FIELD VALUE CONSTRAINTS

4.1. Simplified description

The field value constraints can be of several types. Some fields can only take a unique value, while other fields can pick a value in a range or in an arbitrary set. For a unique possibility, the constraint can be easily performed by checking the received bits one by one along the path in the trellis. However, the task is more difficult for an arbitrary set of possibilities. Indeed, it is then necessary to store the received bits of a field in a register in order to be able to apply the constraints on the whole or a part of the field. Moreover, for improving performance, it is necessary to minimize the constraint description complexity. The set of values is then described as a range of acceptable values and a range of excluded values. In addition, a list of default values can be set, e.g., default values are defined for the location when the GPS is not available.

4.2. Bit order

The AIS messages are transmitted with the bits flipped by octet. More precisely, octets are sent in the correct order, but the bits are sent by starting from the last one. Given that the field lengths are not multiple of octets, the message fields can

be interleaved at the receiver as shown in Fig. 2. In order to store the values of all crossing fields along the paths on the trellis, several registers are defined. Since there are at most 4 fields transmitted simultaneously in the AIS system, 4 registers are used in this paper.

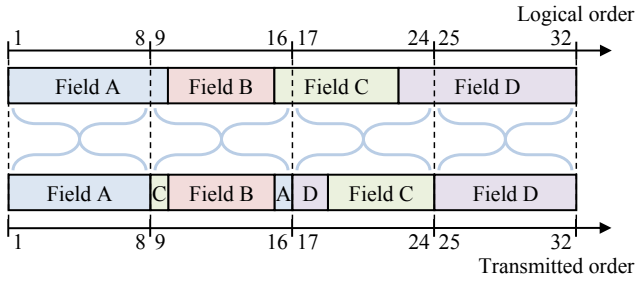

Fig. 2. Interleaved fields.

5. TRANSITION INSTRUCTIONS

5.1. Instruction composition

The instruction associated with each information bit of the messages and executed on a transition in the trellis is composed of elementary operators and corresponding operands. These operators can read the current bit, store it in a register or verify the validity of the value of a register. An operator can modify the metric of the current transition to disable its path on the trellis. Table 1 shows the list of the operands and Table 2 lists the operators and the operands they require.

Several operators can be defined simultaneously on one same instruction in order to be performed together on one transition in the trellis. This allows one, for example, to copy the current bit on a register (*OB* or *OS*) and to check the resulting value of the register with the constraints (*OL*, *OU*, *OE* and/or *OD*).

Table 1. List of operands

Name	ID	Description
Register number	*R*	Register number to read or to write on.
Bit number	*B*	Bit number in the register to write on.
Lower bound	*L*	Lower validity bound.
Upper bound	*U*	Upper validity bound.
Exclusion	*E*	Exclusion interval.
Default values	*D*	List of valid values.

5.2. Bit Stuffing

Since the instructions are executed on transitions carrying information bits, it is necessary to know when a stuffing bit has been received and the number of the current information bit into the message. The state variables S and P represent the number of stuffing bits received before reaching the state and the number of bits 1 received just before the state, respectively. They can be used to determine if the current bit is an information bit and its location in the message.

6. CONSTRAINT CONVERSION

In order to apply the presented algorithm, the high level constraints must be converted into a list of instructions. This section first presents the basic conversion method, and then gives some refinements that enhance performance.

6.1. Basic method

The constraints are converted differently depending on their type. The first type is when a field can take only one value. In this case, an instruction sequence composed of direct test operators (*O0* and *O1*) is generated. These instructions will ensure that the value of the field is correct during the reception of the whole field.

All the other types of constraints are treated in the same way. They need to store the received bits in a register and after the last bit of the field has been received, the value of the register is checked. Thus, a sequence of instructions is made to store the bits in the register. This sequence is composed of copy operators *OB* and *OS* with an additional *OR* in the first instruction to initialize the register. In order to ensure the constraint, some verification operators can be added to the last instruction (*OL*, *OU*, *OE* and *OD*).

6.2. Refinements

The performance of a path selection algorithm in a trellis is largely dependent on the ability to eliminate invalid paths as early as possible. Indeed, as these invalid paths propagate through the trellis, they evict other paths that are potentially valid and do not contain any error. It is therefore important to optimize the algorithm. Indeed, the basic method will eliminate invalid paths only while receiving the last bit of the field for which the constraint is not a single possible value. The proposed refinements are the following:

- If according to the constraint, the current bit and all more significant bits can take only one value, then it is possible to use the operator *O0* or *O1* to test directly the bit value.

- After receiving the last bit of the transmitted field for the current byte, it is possible to check the validity of the register value. Indeed, the register contains at this time all bits more significant than the current bit. Thus it can be compared to the most significant bits of the bounds.

Table 2. List of operators and their required operands

Name	ID	*R*	*B*	*L*	*U*	*E*	*D*	Description
Bit 0	*O0*							Disables the transition if the bit it carries is not 0.
Bit 1	*O1*							Disables the transition if the bit it carries is not 1.
Read bit	*OB*	•	•					Copies the bit carried by the transition in the *B*th bit of the register *R*.
Read sign	*OS*	•	•					Copies the bit carried by the transition in the *B*th bit of the register *R* as a sign bit.
Lower bound	*OL*	•		•				Disables the transition if the value of the register *R* is lower than the bound *L*.
Upper bound	*OU*	•			•			Disables the transition if the value of the register *R* is greater than the bound *U*.
Exclusion	*OE*	•				•		Disables the transition if the value of the register *R* is between the bounds *E*.
Default values	*OD*	•					•	The transition cannot be disabled if the value of the register *R* is in the list of the default values *D*.
Reset register	*OR*	•						The value of the register *R* is reinitialized to 0. This operator is the first to be performed on an instruction.

- If for all bits more significant than the current bit, both bounds of the constraint are the same, then the register validity check can be made at the current bit for all less significant bits.

6.3. Register selection

The fields in the AIS messages can be received interlaced. Thus it is necessary to use several registers in order to store the value of all fields received simultaneously. The choice of a register to use for a given field is made depending of the registers selected for the surrounding fields. The register must not be used for other fields in the first and the last octet containing the field.

7. SIMULATIONS

This section presents some simulation results obtained for the AIS system with the realistic constraints shown in Table 3. The value of the fields must be between the bounds or be equal to one of the default values presented in Table 3. These constraints are defined according to the AIS recommendation and the physics which limits the rate of turn, the speed over ground and the position around the receiver. The repetition of the messages is also considered. In this case, the ship ID and the position accuracy are known. Moreover, the position is around that of the previous message and is limited by the maximum speed of the vessel. In the simulations, the messages are generated in agreement with the constraints. They are composed of 168 information bits concatenated with a 16-bit CRC. The stuffing bits are then inserted according to the AIS recommendation. The frame is encoded with NRZI, and modulated in GMSK with a bandwidth-bit-time product parameter $BT = 0.4$. The generator polynomial for CRC computation is $G(x) = x^{16} + x^{12} + x^5 + 1$ (specified by the AIS recommendation). An additive white Gaussian noise (AWGN) channel is used as in satellite communications. In this paper, one assumes perfect carrier and timing recoveries.

Table 3. Constraints defined for the simulations.

Field name	Bounds	Default
Message ID	$1 \rightarrow 3$	
Repeat indicator		0
Navigational status	$0 \rightarrow 10$	15
Rate of turn	$-37 \rightarrow 37$	$-128, \pm 127$
Speed over ground	$0 \rightarrow 511$	1023
Longitude	$-10^\circ \rightarrow 35^\circ$	181°
Latitude	$-10^\circ \rightarrow 35^\circ$	91°
Course over ground	$0 \rightarrow 3600$	
True heading	$0 \rightarrow 359$	511
Special manoeuvre	$0 \rightarrow 2$	
Spare		0

Fig. 3 shows the performance of the proposed receiver compared to the conventional AIS receiver and to the receiver designed in [2], which did not take into account these new constraints. The performances of the new receiver are illustrated in two cases: when the received message is the first

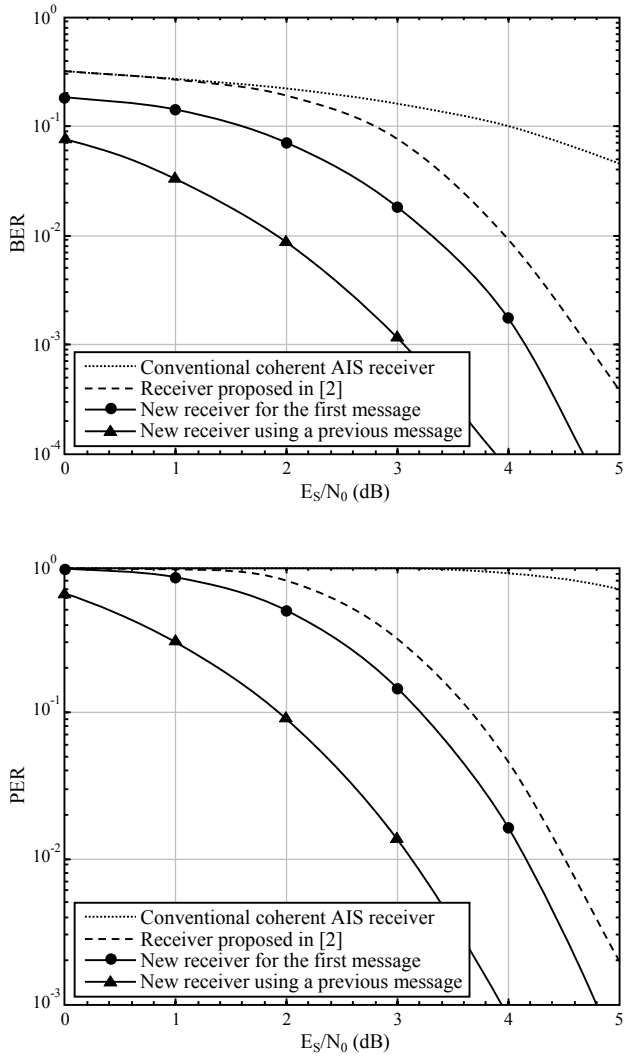

Fig. 3. Comparison in BER and PER between the proposed receiver with and without repetition and the receiver without the constraints.

one, and when there exists a previous message, which implies supplementary constraints. It can be observed that these performances are about 0.5 dB and 2 dB better than those obtained with the receiver without constraint proposed in [2], respectively.

8. CONCLUSION

This paper studied a new error correction strategy using the knowledge of some constraints in the data carried by the messages. Based on a previously proposed corrector [2], the proposed algorithm can handle the bit stuffing of the AIS messages and allows both the CRC and the constraints to be used simultaneously to increase the performance. The corrector can handle the bit stuffing procedure and more complex constraints when compared to the C-VA. The proposed simulations showed than using a larger number of constraints improves the decoding performance.

9. REFERENCES

[1] Recommendation ITU-R M.1371, "Technical characteristics for a universal automatic identification system using time division multiple access in the VHF maritime mobile band." ITU, 2001.

[2] R. Prévost, M. Coulon, D. Bonacci, J. LeMaitre, J.-P. Millerioux, and J.-Y. Tourneret, "CRC-assisted error correction in a trellis coded system with bit stuffing," in *Proc. IEEE Workshop on Stat. Signal Processing*, Nice, France, June 2011, pp. 381–385.

[3] R. Wang, W. Zhao, and G. B. Giannakis, "CRC-assisted error correction in a convolutionally coded system," *IEEE Trans. Comm.*, vol. 56, no. 11, pp. 1807–1815, Nov. 2008.

[4] C. Marin, Y. Leprovost, M. Kieffer, and P. Duhamel, "Robust MAC-lite and soft header recovery for packetized multimedia transmission," *IEEE Trans. Comm.*, vol. 58, no. 3, pp. 775–784, March 2010.

[5] C. Marin, K. Bouchireb, M. Kieffer, and P. Duhamel, "Joint exploitation of residual source information and MAC layer CRC redundancy for robust video decoding," *IEEE Trans. Wireless Comm.*, vol. 9, no. 7, pp. 2165–2175, July 2010.

[6] R. Prévost, M. Coulon, D. Bonacci, J. LeMaitre, J.-P. Millerioux, and J.-Y. Tourneret, "Interference mitigation and error correction method for AIS signals received by satellite," in *Proc. European Signal Proc. Conf.*, Bucharest, Romania, Aug. 2012, to appear.

[7] L. Cao and C. W. Chen, "A novel product coding and recurrent alternate decoding scheme for image transmission over noisy channels," *IEEE Trans. Comm.*, vol. 51, no. 9, pp. 1426–1431, Sept. 2003.