P. A. Franaszek

Synchronous Bounded Delay Coding for Input Restricted Channels

Consider the class of input restricted channels whose constraints can be modelled by finite state machines. This paper treats the problem of constructing fixed rate (synchronous) codes for such channels subject to a constraint that the coding delay be bounded by a parameter M. M is the maximum number of information symbols required by the coder in choosing a word during transmission.

1. Introduction

In many digital transmission and recording systems, considerations such as spectral shaping, self-timing, and limitations on intersymbol interference require that, before modulation, the data be mapped onto a sequence with special properties. Such suitable sequences often define discrete noiseless channels of the type considered by Shannon [1] in which restrictions are represented by finite-state sequential machines. In practice, a common additional requirement is that the encoding of the binary data onto the channel sequence must be synchronous. By this is meant that the bit per symbol ratio (the coding rate) is constant over each word.

An extensive body of literature exists on such codes; References [2, 3] are survey papers. Although there have been techniques proposed for creating a direct mapping between data and code words [4–7], the majority of channel codes encountered in practice employ table look-up techniques; construction of such codes requires finding a set of code words or paths which correspond to state transitions in the model for the channel constraints. Information to be coded is then associated with these paths [8, 9].

This paper considers questions related to the problem of formulating synchronous codes under the assumptions that (a) the code word choice may be a function of the channel state, and (b) code words are chosen with knowledge of up to M of the next letters to be transmitted. Here M is referred to as the coding delay. Such codes include, as special cases, the fixed and variable length (FL and VL) codes of References [9–11], as well as the future dependent (FD) block codes of [12].

The FL and VL codes of References [9–11] have the property that information to be represented by a word is, given the channel state, sufficient to determine this word. Knowledge of all the next M known letters to be transmitted is not used unless the code word is to represent these source letters. There is no look-ahead or future dependency. A consequence is that the number of code words available from each coding state must be sufficient to satisfy the Kraft inequality.*

Recently it was shown [12] that, for the special case of block codes, this condition is in fact not necessary; if future dependency is permitted, the Kraft inequality for the required number of code word choices may be replaced by a weaker set of relations on the number of paths linking such states. These permit the construction of codes for rates and channel constraints which admit no conventional codes of finite length. Although this is true only for certain special cases and for rates equal to the channel

*The Kraft inequality is given by

$$1 \le \sum_{j=1}^{\infty} n_j 2^{-j\alpha},$$

where $j\alpha$ is the number of source bits represented by a word of length j and n_j is the number of words of this length.

capacity, a practical implication which holds more generally is that the techniques may result in an appreciable reduction of the word (or dependency) length required to obtain a given rate. Since implementation is generally via hard wired logic or table look-up, this tends to yield a significant decrease in coder complexity, as well as reducing error propagation.

This paper extends the earlier block coding result [12] to the general case of variable word length. A recursive algorithm, based on dynamic programming, is developed for finding channel states which are candidates for a set of coding states. The existence of a set of such states is a necessary condition for the possibility of constructing a code.

The following is a synopsis of the paper. Section 2 reviews methods for generating paths between the channel states. Section 3 contains a derivation of a set of relations between the number of coding paths connecting the channel states which is necessary for the existence of a code. In the special case where the number of paths from each state satisfies the Kraft inequality, the associated code is a conventional one of fixed or variable length. Section 4 describes an algorithm for finding a set of states eligible for coding. The existence of such a set is a necessary but not sufficient condition for the existence of a code. Section 5 presents an example, and Section 6 summarizes the results.

2. Coding paths and channel constraints

Let $S = \{\sigma_i\}$, $i = 1, 2, \cdots, E$, denote the states for the channel constraint model. To each $\sigma_i \in S$, there corresponds a set of allowable channel symbols $\{V_k\}_i$. The transmission of a symbol takes the channel to a new state which is a function of the previous state and the transmitted symbol.

It is convenient to define a channel skeleton transition matrix as follows:

$$\mathbf{D} = \{d_{ij}\}, \tag{1}$$

where $d_{ij} \triangleq$ the number of distinct paths of length one (*i.e.*, channel symbols) which take the channel from σ_i to σ_j.

Let $d^n_{ij} \triangleq$ the number of paths of length n symbols which take the channel from σ_i to σ_j. Then

$$d^n_{ij} = [\mathbf{D}^n]_{ij}. \tag{2}$$

The channel capacity, defined as the maximum bit per symbol rate permitted by the channel constraints, is given [1] by the base-two logarithm of the largest real root of

$$\det\,[d_{ij}Z^{-1} - \delta_{ij}] = 0, \tag{3}$$

where δ_{ij} is the Kronecker delta and Z a dummy variable.

Symbols associated with paths of length W may be obtained from the Wth power of the channel transition matrix

$$\mathbf{A} = \{a_{ij}\}, \tag{4}$$

where a_{ij} represents the disjunction, +, of channel symbols which cause a transition from σ_i to σ_j. If no such symbol exists, then $a_{ij} = \varnothing$, the null sequence. Powers of $\mathbf{A}$ are formed by the operations of disjunction, +, and concatenation. The concatenation of $\varnothing$ with any symbol results in $\varnothing$.

3. Necessary conditions

Let L_j, $j = 1, 2, \cdots$ be the sequence of letters to be transmitted. It is assumed that each letter L_j is drawn from an alphabet B, where $\alpha = \log_2 |B|$. That is, each letter represents α bits. Let W denote the *basic word length*. That is, code words of length NW, $N \le M$, represent N source letters. The coding rate $R = \alpha/W$.

Let $\{\sigma_i\}^c$ be a set of *coding* or *terminal states*. These are the states entered at the end of code words.

- *Definition*

A synchronous, instantaneously decodable bounded delay code $C(S, M, R)$ is a mapping

$$C(\sigma_i;\, L_j, L_{j+1}, L_{j+2}, \cdots, L_{j+M-1}) \rightarrow \omega \tag{5}$$

of letters L onto paths between the channel states S, where ω is a sequence of NW channel symbols associated with the next N letters to be transmitted, $L_j, L_{j+1}, \cdots, L_{j+N-1}$, when the channel occupies $\sigma_i \in \{\sigma_i\}^c$. The mapping is such that $L_j, L_{j+1}, \cdots, L_{j+N}$ can be recovered given σ_i and ω, and which is defined for those combinations of states and source letters which occur during transmission. □

It should be noted that the above definition does not require that the mapping C be valid for any combination of coding state and letters to be transmitted. That is, the transmission of a particular sequence of letters may require that the channel state be in a proper subset of $\{\sigma_i\}^c$. This point is discussed further below.

- *Definition*

An *independent path set* of length NW (which will usually be termed an independent path or IP for short) is a path or set of paths of length NW from a given terminal state which is capable of representing one particular sequence of N source letters, with no constraints on the letters that follow. □

Note that an IP of length N is equivalent to 2^α independent paths of length $N + 1$.

• *Proposition 1*

Suppose state σ_i is entered at the end of a word ω of length NW at time t_{j+N-1}. Then σ_i must have leading from it an integer number of independent paths of length $(M - N)W$.

Proof

The word ω represents the first N of the M source letters $(L_j, L_{j+1}, \cdots, L_{j+M-1})$ known at the time t_j. Clearly σ_i must have associated with it enough paths so that the mapping $C(\sigma_i; L_{j+N}, L_{j+N+1}, \cdots, L_{j+M+N-1})$ is defined for any $(L_{j+M}, \cdots, L_{j+M+N-1})$. □

• *Definition*

Let ϕ_i be the *weight* of state σ_i, where

$$\phi_i = \sum_{N=0}^{n} Z_N 2^{-\alpha N}, \tag{6}$$

and Z_N is the number of distinct independent paths of length NW leading from σ_i. □

An independent path of length NW is said to be of weight $2^{-\alpha N}$. Note that if each source sequence corresponds to a single code sequence (*i.e.*, the code is not redundant), the weight ϕ_i corresponds to the number of source sequences that can be encoded from this state.

Consider a code word ω of length NW that terminates in state σ_i. From Proposition 1, the code words available from σ_i to follow ω must comprise an integer number of independent paths of length $(M - N)$. Suppose the code is such that the code paths to be used from σ_i are independent of the way this state is entered. Then the code will be termed *memoryless*. To illustrate this notion, suppose σ_i has independent paths corresponding to the source sequences 01 and 00. A code with memory might be constructed such that if σ_i is entered via ω_0 from σ_j, then the next two source bits to be transmitted are necessarily 00.

• *Proposition 2*

Suppose, for a given memoryless code, there exists a path ω of length NW leading from σ_i to σ_j. Then this path is of weight $2^{-\alpha N}\phi_j$.

Proof

State σ_j has $\phi_j = r2^{-\alpha(M-N)}$ independent paths of length $(M - N)W$ for some integer r (from Proposition 1). Then ω must be capable of representing a particular sequence of N letters, followed by any of r sequences of $(M - N)$ letters, followed by any source letters. Thus, ω followed by the set of paths available from σ_j is equivalent to r independent paths of length MW and is of weight $r2^{-\alpha M} = 2^{-\alpha N}\phi_j$. □

If the code is not memoryless, a path of the type described in Proposition 2 has weight no greater than $2^{-\alpha N}\phi_j$.

Let $C(S, M, R)$ be a code and $\{P^r_{ij}(N)\}$, $N = 1, 2, \cdots, M$, $r = 1, 2, \cdots$ be the set of code paths of length NW starting at τ_i and terminating at τ_j. The superscript r is necessary since there may be several paths of length NW leading from σ_i to σ_j. Let $|\{P^r_{ij}(N)\}|$ denote the number of such distinct paths.

• *Proposition 3*

A necessary condition for the existence of a memoryless $C(S, M, R)$ code is that the set of equations

$$\phi_i = \sum_{j,r,N} |\{P^r_{ij}(N)\}|\, 2^{-\alpha N}\phi_j \tag{7}$$

has a set of solutions such that

a. The ϕ_i are integer multiples of $2^{-\alpha(M-N)}$, where NW is the maximum length of the code paths terminating in ϕ_i.
b. Not all the ϕ_i are zero.
c. $\phi_i \le 1$.

Proof

Conditions (a) and (b) follow immediately from the definition of ϕ_i and Propositions 1 and 2. Condition (c) is necessary if there are to be no redundant paths in the code, since $\phi_i = 1$ implies that sufficient paths exist from σ_i for the mapping $C(\sigma_i; L_j, L_{j+1}, \cdots, L_{j+M-1})$ to be valid for any $L_j, \cdots, L_{j+M-1}$. □

Consider the special case where all nonzero ϕ_i are equal to 1. Then

$$\phi_i = \sum_{j,r,N} |\{P^r_{ij}(N)\}|\, 2^{-\alpha N} = 1, \tag{8}$$

which satisfies the Kraft inequality

$$\sum_{j,r,N} |\{P^r_{ij}(N)\}|\, 2^{-\alpha N} \ge 1. \tag{9}$$

This case corresponds to the variable length codes of [10]. Here each path $P^r_{ij}(N)$ can be assigned a distinct source sequence of length N.

If some coding states are of fractional weight, then two additional conditions must be satisfied in order for memoryless instantaneously decodable code to exist [12]:

d. Each distinct sequence of N source letters assigned (*i.e.*, for which the mapping C is valid) to a state σ_i and which is associated with a path $P^r_{ij}(N)$ must be the only such sequence associated with $P^r_{ij}(N)$.

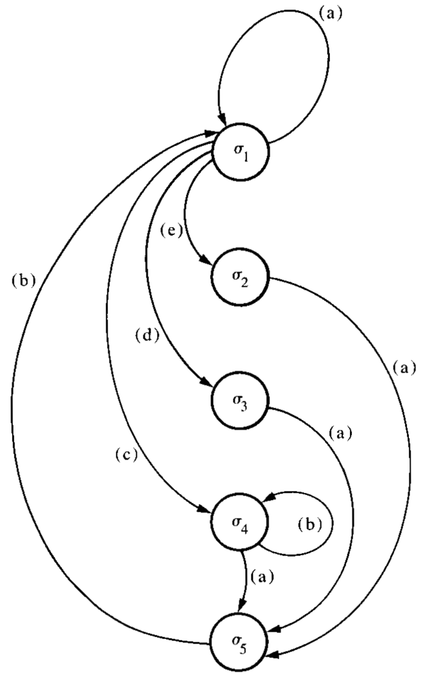

Figure 1 Model of a constrained channel.

e. The sequences of letters assigned to a coding state σ_i are independent of the path used to reach this state.

If a code is not required to be memoryless, then clearly

$$\phi_i \le \sum_{j,r,N} |\{P^r_{ij}(N)\}|\ 2^{-\alpha N}\phi_j, \tag{10}$$

since the number of independent paths used from a state may be a function of the way this state was entered. A state σ_i entered by a path of length NW must have a subset of independent paths whose combined weight is an integer multiple of $2^{-\alpha(M-N)}$.

4. Candidate coding states

If a conventional variable length code is to be constructed (*i.e.*, one where all terminal state weights are 1), then [10] describes an algorithm, based on dynamic programming, for finding a set of feasible coding states. In the terminology of this paper, this is a set of coding states each of which is of weight no smaller than one, termed the *principal states*.

This section describes a generalization of the above algorithm. The result of performing the computation is a set of states $\{\sigma_i\}$ which are candidate coding states. If this set is empty, no code can be constructed with the given constraints of delay M and rate R. Otherwise, a subset of the paths connecting these states might satisfy the conditions specified in the previous section for coding either with or without memory.

In the case of conventional variable length codes, the set of paths leading from each principal state σ_i and terminating at other principal states satisfies the Kraft inequality. It is not difficult to show that, given such a set of paths, a subset (of weight one) exists which can be used as code words. The analogous property does not hold for the case of fractional weights. If a set of paths exists with weight greater than $r2^{-\alpha N}$ (with r an integer), this does not necessarily imply that a subset can be found with weight exactly $r2^{-\alpha N}$.

Suppose $\{\phi^*_j\}$, $\sigma_j \in S$, is an initial set of approximations for the state weights. A procedure is described below which maximizes the weight of a state σ_i, ϕ_i, in terms of the $\{\phi^*_j\}$. The technique is based on the observation that a path is not worth extending (in terms of the contribution to ϕ_i) past a given state at depth N (times the basic code word length W) unless there are enough path extensions to states of sufficient weight. States beyond which paths of length NW are not worth extending will be called *end states* for depth NW. The set of paths leading from σ_i to such states then yields a maximum value for ϕ_i.

Let $T(\sigma_i, D)$ be the set of end states for paths of length DW leading from σ_i. Initially $T(\sigma_i, M) = S$ and $T(\sigma_i, D)$ is empty for $D < M$.

Let $\{P^r_{ik}(D)\ |\sigma_j, F\}$ be the set of paths of length DW from σ_i to σ_k which enter σ_j at depth FW and which do not enter a member of $T(\sigma_i, Q)$ for $Q < D$ at depth QW.

Let

$$r_j(F) = 2^{-(M-F)\alpha}\ \text{floor}\ [2^{(M-F)\alpha}\phi^*_j]. \tag{11}$$

Note that the quantity $r_j(F)$ is an upper bound for the usable weight (from Proposition 2) of state ϕ_i for a path of length F terminating there. It is not necessarily the maximum usable weight since there may not exist a subset of paths from σ_j which actually yield this weight.

Given that end state sets $T_i(\sigma_i, N)$ have been obtained for $N > F$, a state σ_j is assigned to $T(\sigma_i, F)$ if

$$2^{-\alpha F}r_j(F)|\{P^r_{ij}(F)\}| \ge \sum_{D=F+1}^{M} \sum_{\sigma_k \in T(\sigma_i, D)} 2^{-\alpha D}|\{P^r_{ik}(D)|\sigma_j, F\}|r_k(D), \tag{12}$$

where $|\{x\}|$ is the cardinality of $\{x\}$, and $\{P^r_{ij}(F)\}$ is the set of paths of length FW which lead from σ_i to σ_k.

Once the end states have been determined for $D = 1, 2, \cdots, M$, an upper bound for ϕ_i can be obtained in terms of the initial $\{\phi^*_j\}$:

$$\phi_i \leq \sum_{D=1}^{M} \sum_{\sigma_k \in T(\sigma_i, D)} 2^{-\alpha D} \,|\{P^r_{ik}(D)|\sigma_k, D\}| r_k(D), \tag{13}$$

subject to the condition that $\phi_i \leq 1$.

The following is an algorithm for finding a set of candidate coding states.

• *Algorithm 1*

```
Initially:
  Flag = 1,
φ*_i = 1, σ_i ∈ S
Start:
  If Flag = 1, then do;
    Flag ← 0;
    Do for σ_i ∈ S, i = 1, 2, · · ·,
    (a) Maximize the weight φ_i in terms of the {φ*_i}.
    (b) If φ_i ≠ φ*_i, then do;
φ*_i ← φ_i
       Flag ← 1;
    END;
  END;
END;
If Flag = 1, go to start;
END; □
```

The resulting set $\{\phi^*_i\}$ are upper bounds for the actual state weights. If all are zero, clearly no code can be constructed within the given constraints. Otherwise, a set of paths may exist which yield solutions to Eqs. (7).

5. Example

An example of code generation is given in this section. The constraints are those shown in Fig. 1. The capacity of this channel is exactly one bit per symbol.

An attempt will be made to construct a memoryless $R = 1$ code with $M = 3$.

Initially (Algorithm 1), $\phi^*_i = 1$, $i = 1, 2, \cdots, 5$. In the first pass of the algorithm $\phi^*_1 = 1$. (There are four paths of length 1 from σ_1, each terminating in states whose current estimated weight is 1.) Similarly, $\phi^*_2 = \phi^*_3 = \phi^*_5 = 1/2$, $\phi^*_4 = 1$. Several of the ϕ^*_i have been changed, so that a second pass is necessary.

The second pass yields $\phi^*_1 = 1$. It should be noted that paths of length 3 can only terminate in states of weight 1. Thus ϕ^*_2 is the largest of (2/8, 1/4, 1/4) depending on the length of the paths chosen. Thus, $\phi^*_2 = 1/4$. Similarly, one obtains $\phi^*_3 = 1/4$, $\phi^*_4 = 3/4$, and $\phi^*_5 = 1/2$.

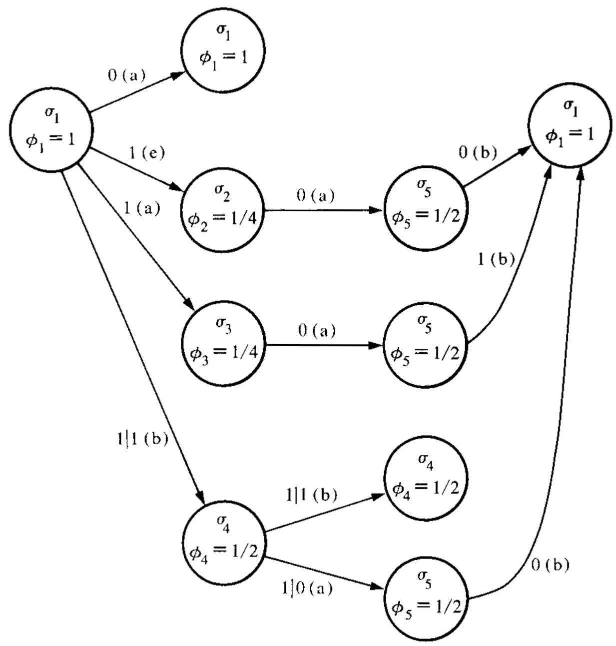

Figure 2 A coding trellis. Letters in parentheses are channel symbols.

Table 1 Source letter assignments.

	Source letters	*Code word*	*New state*
σ_1	0	a	σ_1
	101	eab	σ_1
	101	dab	σ_1
	1|1	c	σ_4
σ_4	1|1	b	σ_4
	10	ab	σ_1

The third pass yields

$$\phi^*_1 = 1,$$
$$\phi^*_2 = \phi^*_3 = 1/4,$$
$$\phi^*_4 = 1/2,$$
$$\phi^*_5 = 1/2. \tag{14}$$

The above value of ϕ^*_4 is obtained from

$$\phi^*_4 \leq 1/2[\phi^*_4 + \phi^*_2] = 5/8. \tag{15}$$

But ϕ^*_4 must be an integer multiple of 1/4. Thus, $\phi^*_4 = 1/2$. A fourth path yields the source values for the ϕ_i, so that these are final. These values satisfy the equation

$$\phi_1 = 2(\phi_1 + \phi_2 + \phi_3 + \phi_4),$$
$$\phi_2 = 2^{-1}(\phi_5),$$
$$\phi_3 = 2^{-1}(\phi_5),$$
$$\phi_4 = 2^{-1}(\phi_4 + \phi_5). \quad (16)$$

Table 1 is an assignment of source letters to the paths. The notation $1 \mid 1$ denotes that the source letter 1 is to be transmitted given knowledge the next source letter is a 1. Note that paths that enter states of weight 1/4 at depth 1 have three letters specified. Figure 2 shows a coding trellis.

6. Summary

A set of necessary conditions were derived for the existence of instantaneously decodable, bounded delay codes for input restricted channels. An algorithm based on dynamic programming was described for bounding the set of state weights.

References

1. C. E. Shannon, "A Mathematical Theory of Communication," *Bell Syst. Tech. J.* **47,** 143-157 (1948).
2. H. Kobayashi, "A Survey of Coding Schemes for Transmission or Recording of Digital Data," *IEEE Trans. Commun. Technol.* **COM-19,** 1087-1100 (1971).
3. P. A. Franaszek, "Noiseless Coding for Signal Design," *Proceedings of the Ninth Allerton Conference*, University of Illinois, Urbana, October 1971.
4. W. H. Kautz, "Fibonacci Codes for Synchronization Control," *IEEE Trans. Info. Theory* **IT-11,** 284-292 (1965).
5. D. T. Tang, "Run Length Limited Codes," *1969 IEEE International Symposium on Information Theory*, IEEE Inc., New York, 1969.
6. D. T. Tang and L. R. Bahl, "Block Codes for a Class of Constrained Noiseless Channels," *Info. Control* **17,** 436-461 (1970).
7. A. M. Patel, "Zero-Modulation Encoding in Magnetic Recording," *IBM J. Res. Develop.* **19,** 366-378 (1975).
8. C. V. Freiman and A. D. Wyner, "Optimum Block Codes for Noiseless Channels," *Info. Control* **7,** 398-415 (1964).
9. P. A. Franaszek, "Sequence-state Coding for Digital Transmission," *Bell Syst. Tech. J.* **47,** 143-157 (1968).
10. P. A. Franaszek, "On Synchronous Variable-Length Coding for Discrete Noiseless Channels," *Info. Control* **15,** 155-164 (1969).
11. P. A. Franaszek, "Sequence-state Methods for Run-length-limited Coding," *IBM J. Res. Develop.* **14,** 376-383 (1970).
12. P. A. Franaszek, "On Future-Dependent Block Coding for Input-Restricted Channels," *IBM J. Res. Develop.* **23,** 75-81 (1979).

Received February 9, 1979; revised July 6, 1979

The author is located at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598.