Mapping of Trellises Associated with General Encoders onto High-Performance VLSI Architectures

MONTSERRAT BÓO, FRANCISCO ARGUELLO, JAVIER D. BRUGUERA
Department of Electronics, University of Santiago de Compostela, Spain

EMILIO L. ZAPATA
Department of Computer Architecture, University of Málaga, Spain

Received January 2, 1996; Revised October 15, 1996

Abstract. A rate $1/n$ binary generic convolutional encoder is a shift-register circuit where the inputs $u_k$ are information bits and the outputs $y_k$ are blocks of $n$ bits generated as linear combinations on the appropriate shift register contents. The decoding of the outputs of a convolutional encoder can be carried out by the well-known Viterbi algorithm. The communication pattern of the Viterbi Algorithm is given as a graph, called trellis, associated to the state diagram of the corresponding encoder. In this paper we present a methodology that permits the efficient mapping of the Viterbi algorithm onto a column of an arbitrary number of processors. This is done through the representation of the data flow by using mathematical operators which present an immediate hardware projection. A single operator string has been obtained to represent a generic encoder through the study of the data flow of free-forward encoders and feed-back encoders. The formal model developed is employed for the partitioning of the computations among an arbitrary number of processors in such a way that the data are restructured optimizing the use of the processors and the communications. As a result, we obtain a highly regular and modular architecture suitable for VLSI implementation.

1. Introduction

There are many algorithms whose communication pattern is given as a task graph (trellis), where nodes denote processes and edges denote communications among processes. The task of projecting these algorithms onto hardware systems can be modeled as a graph embedding problem. Among these algorithms we point out the Viterbi algorithm [1], which has been widely applied to many decoding and estimation applications in communications and signal processing. It can be used to perform maximum likelihood decoding for convolutional codes, or maximum likelihood esti-

*This work was supported by the Ministry of Education and Science (CICYT) of Spain under contract TIC92-0942 and Xunta de Galicia XUGA-20605B95

Trellis Coded Modulation has evolved over the past decade as a combined coding and modulation technique for digital transmission over band-limited channels. Trellis Coded Quantization was recently introduced as an effective scheme for encoding memoryless sources with low to moderate complexity. In fact the motivation for TCQ comes from Ungerboeck's formulation of TCM. In both cases the interconnection pattern among processors that is traditionally employed may be generated by means of a convolutional encoder with feedback. For a given data sequence, the Viterbi algorithm is then used to find the minimum mean square error (MSE) path through the trellis. The central unit of a Viterbi decoder is a data-dependent feedback loop
which performs an add-compare-select (ACS) operation.

There are several alternatives for the implementation of the Viterbi algorithm in a VLSI architecture: the state-serial [5] strategy in which one (or a few) processor is used for the computation of the whole trellis or parallel processing [6, 7, 8] in which there are as many ACSs as states. However, the good results obtained for the algorithms with trellises with a high number of states makes the fully parallel architectures very costly in terms of hardware and state-serial architectures too slow for most applications. For this reason a need arises for the use of intermediate solutions in which the number of ACSs can be preset according to the speed requirements and the availability of area, so that each ACS is shared by more than one state of the trellis.

A projection methodology based on the use of intermediate solutions is presented in [9, 10, 11, 12, 13]. This results in a continuous means of trading speed for area, in which we find a design solution for particular area or speed constraints. However, the mapping of the states of the trellis onto the ACS network is based on matrix permutation techniques where the working matrix is heuristic. Another type of projection methodology consists in the use of a mathematical model that permits representing in a simple way the data flow among states of the trellis as well as its mapping onto a processor column. Mathematical models of this type have been developed for data flows associated with the FFT [14, 15], tridiagonal systems [16] and free-forward convolutional codes [17].

However, these methodologies can only be applied to convolutional encoders with no feedback, and consequently, their extension to encoders with feedback is important. The use of a system with feedback implies the study of new data flows, and, in particular, each interconnection has different trellises associated to it. The extension of the mathematical model to these systems in principle implies the association of a projection model to each data flow and, consequently, to each interconnectivity of the encoder.

In this work we present a general methodology for carrying out the partitioning and mapping of a trellis associated with a general encoder with feedback onto an array of processors of arbitrary size. The resulting architecture is a processor column of arbitrary size in which the data are recirculated by means of the use of local routing in each processor and global routing among processors. These two types of routing permit the recirculation of the data so that they are presented to each processor in the correct order and instant of time for processing the data of the next state of the trellis. The projection methodology is based on the use of mathematical operators that permit the description of the data flow and which have a direct hardware projection. According to this, each encoder, and thus, each associated trellis, may be represented by means of an operator string. We have combined all these operator sequences into a single one so that the designer can project, in a simple and deterministic way, the data flow associated with any encoder onto an ACS array of arbitrary size.

2. Mapping of trellises of encoders without feedback: Previous work

In this section we present the mathematical model for the projection of free-forward convolutional codes with no feedback onto area efficient architectures that has been recently developed [17] and which we will use as a base for the projection of encoders with feedback. A free-forward encoder consists of a shift register with \( n + 1 \) stages and \( k \) binary function generators. We denote as state the content of the \( n \) most significant bits of the shift register and \( N \) as the number of states of the system \( (N - 2^n) \). The binary data input is shifted along the shift register by one bit each instant of time. Figure 1(a) shows an example of an encoder for \( n=2 \), \( k=2 \). It is a synchronous state machine with one input \( x_{in} \), two outputs \( x_1, x_2 \) and \( N = 2^2 \) states.

From the state diagram of the encoder it is possible to obtain the trellis. It is a two dimensional representation in which we represent the states in the vertical direction and the temporal transitions are marked in the horizontal direction. The states in an instant of time are connected to those in instant \( t+1 \) by branches that specify possible state transitions. However, as the transition scheme is repeated in time, it is enough to specify the transitions between instants \( t \) and \( t+1 \). Figure 1(b) shows a trellis diagram of the convolutional encoder of figure 1(a) and in figure 1(c) displays the same trellis but rearranging the states so that the butterfly structure of any trellis of a convolutional code is evident. From now on we will represent state \( [i] \) in instant \( t \) as \( [i]_{(t)} \) although the temporal subscript will be eliminated whenever its specification is not necessary.