ACKNOWLEDGMENT

The author is grateful to Prof. J. Moore who suggested to look at the Baum–Welch and EM algorithms, and to V. Krishnamurthy for interesting conversations. The hospitality of the Systems Engineering Department, Research School of Physical Sciences, Australian National University is also acknowledged.

REFERENCES

[1] P. Brémaud, *Point Processes and Queues.* New York: Springer-Verlag, 1981.
[2] D. Clements and B. D. O. Anderson, "A non-linear fixed lag smoother for finite-state Markov processes," *IEEE Trans. Inform. Theory,* vol. IT-21, p. 446–452, 1975.
[3] A. Dembo and O. Zeitouni, "Parameter estimation of partially observed continuous time stochastic processes via the EM algorithm," *Stoch. Proc. Applic.,* vol. 23, pp. 91–113, 1986. Erratum vol. 31, pp. 167–169, 1989.
[4] R. J. Elliott, *Stochastic Calculus and Applications,* in Applications of Mathematics, vol. 18. New York: Springer-Verlag, 1982.
[5] ——, "The nonlinear filtering equations," in *Lecture Notes in Control and Information Sciences 43.* New York: Springer-Verlag, 1982, pp. 168–178.
[6] W. M. Wonham, "Some applications of stochastic differential equations to optimal non-linear filtering," *SIAM J. Contr.,* vol. 2, pp. 347–369, 1965.
[7] Y. C. Yao, "Estimation of noisy telegraph process: Nonlinear filtering versus nonlinear smoothing," *IEEE Trans. Inform. Theory,* vol. IT-31, pp. 444–446, May 1985.
[8] O. Zeitouni and A. Dembo, "Exact filters for the estimation of the number of transitions of finite-state continuous time Markov processes," *IEEE Trans. Inform. Theory,* vol. 34, pp. 890–893, July 1988.
[9] R. J. Elliott, "Exact adaptive filters for Markov chains observed in Gaussian noise," Univ. of Alberta, preprint.

Computing m DFT's Over GF(q) with One DFT Over GF(q^m)

Jonathan Hong and Martin Vetterli

***Abstract*—Over the field of complex numbers, it is well-known that if the input is real then it is possible to compute 2 real DFT's with one complex DFT. We extend the result to finite fields and show how to compute m DFT's over GF(q) with one DFT over GF(q^m).**

***Index Terms*—Finite fields, DFT, complexity.**

I. INTRODUCTION

The discrete Fourier transform and its inverse are defined as

$$X_k = \sum_{n=0}^{N-1} x_n W_N^{nk}, \tag{1}$$

$$x_k = N^{-1} \sum_{n=0}^{N-1} X_n W_N^{-nk}. \tag{2}$$

Manuscript received February 5, 1991; revised March 1, 1992. This work was supported in part by the National Science Foundation under Grants CDR-84-21402 and MIP-90-14189.

The authors are with the Department of Electrical Engineering and Center for Telecommunications Research, Columbia University, Room 1342 S.W. Mudd Building, 500 W. 120th Street, New York, NY 10027–6699.

IEEE Log Number 9203639.

The definition is valid over any field $\boldsymbol{F}$ provided that W_N, an element of order N, exists in $\boldsymbol{F}$ and N^{-1} exists in $\boldsymbol{F}$.

For $\boldsymbol{F} = \boldsymbol{C}$, the field of complex numbers, it is well known that if the input x_n is real then it is possible to compute two real DFT's with one complex DFT [6], thereby showing that the complexity of a real DFT is approximately half that of a complex DFT. This reduction in complexity is due essentially to the fact that the $\boldsymbol{C}$ is an extension of degree 2 over $\boldsymbol{R}$, i.e., $[\boldsymbol{C}:\boldsymbol{R}] = 2$. In this correspondence, we will show that this result extends to the case where $\boldsymbol{F}$ is a finite field. Specifically, if $\boldsymbol{F}$ is an extension field of $\boldsymbol{K}$ with $[\boldsymbol{F}:\boldsymbol{K}] = m$, then the complexity of the DFT of a $\boldsymbol{F}$-sequence is approximately m times that of a $\boldsymbol{K}$-sequence. We will demonstrate this by showing how to compute the DFT of m $\boldsymbol{K}$-sequences with the DFT of 1 $\boldsymbol{F}$-sequence. The procedure is quite general and will be shown to apply to the real/complex case as well. As is customary, we will assume that the constants in the algorithm are precomputed and thus the cost of their computation excluded from the overall complexity of the algorithm.

II. CONJUGACY RELATIONS

We derive, in this section, the well-known conjugacy relationship that exists among the elements of the DFT of a subfield sequence. The derivation differs (conceptually) from the usual derivation in that it makes use of the automorphism group associated with the fields $\boldsymbol{F}$ and $\boldsymbol{K}$. While it is possible to present our algorithm without reference to the automorphism group; we choose to introduce it in order to keep the derivation as general as possible. The generality eliminates the need to rederive the algorithm for the real and complex case in Section IV.

Let $\boldsymbol{F}$ and $\boldsymbol{K}$ be finite fields such that $\boldsymbol{K} \subset \boldsymbol{F}$, then there are two ways we can view $\boldsymbol{F}$ relative to $\boldsymbol{K}$. First, with respect to the addition operation in $\boldsymbol{F}$, $\boldsymbol{F}$ is a $[\boldsymbol{F}:\boldsymbol{K}]$-dimensional vector space over $\boldsymbol{K}$ (notation: $\boldsymbol{F}_{\boldsymbol{K}}$). Thus if $\boldsymbol{K} = GF(q)$ and $[\boldsymbol{F}:\boldsymbol{K}] = m$, then $\boldsymbol{F} = \mathrm{GF}(q^m)$; from which it follows that if $\{\beta_0, \beta_1, \cdots, \beta_{m-1}\}$ is any basis of $\boldsymbol{F}_{\boldsymbol{K}}$ then any $x \in \boldsymbol{F}$ can be written uniquely in the form

$$x = x_0\beta_0 + x_1\beta_1 + \cdots + x_{m-1}\beta_{m-1}$$

with the coefficients x_i in $\boldsymbol{K}$. Alternatively we can view $\boldsymbol{F}$ as an extension field of $\boldsymbol{K}$. Associated with this field extension is the group $\mathrm{Aut}_{\boldsymbol{K}}\boldsymbol{F}$ which is the group of automorphisms of $\boldsymbol{F}$ that leave $\boldsymbol{K}$ *fixed.* It is well known that this group is cyclic of order $[\boldsymbol{F}:\boldsymbol{K}]$ and has as its generator

$$\psi : a \mapsto a^q,$$

[1]–[3]. In other words, $\mathrm{Aut}_{\boldsymbol{K}}\boldsymbol{F} = \{\varphi_l\}_{l=0}^{m-1}$ where

$$\varphi_l = \psi^l : a \mapsto a^{q^l}. \tag{3}$$

Consider

$$X_k = \sum_{n=0}^{N-1} x_n W_N^{nk}$$

with $x_n \in \boldsymbol{K}$ for every n. Since φ_l is a homomorphism which fixes elements of $\boldsymbol{K}$, we see that

$$\varphi_l(X_k) = \varphi_l\left(\sum_{n=0}^{N-1} x_n W_N^{nk}\right) = \sum_{n=0}^{N-1} \varphi_l\left(x_n W_N^{nk}\right)$$
$$= \sum_{n=0}^{N-1} \varphi_l(x_n)\varphi_l\left(W_N^{nk}\right) = \sum_{n=0}^{N-1} x_n \varphi_l\left(W_N^{nk}\right),$$

therefore,

$$X_k^{q^l} = \sum_{n=0}^{N-1} x_n W_N^{nkq^l} = X_{kq^l}. \tag{4}$$

Equation (4) is the familiar conjugacy relation [4], [5]. We call $X_k^{q^l}$ a *conjugate* of X_k and $\left\{X_k^{q^l}\right\}_{l=0}^{m-1}$ the *conjugacy class* of X_k.

Define

$$X_i \sim X_j, \qquad \text{if } j = iq^l, \qquad \text{for some } l \in \boldsymbol{Z}; \tag{5}$$

then $\sim$ is an equivalence relation on $\{X_k\}$ with $\left\{X_{kq^l}\right\}_{l=0}^{m-1}$ the *equivalence class* of X_k. Equation (4) says that when $x_n \in \boldsymbol{K}$, then the conjugacy class of X_k is the same as the equivalence class of X_k, i.e.,

$$\begin{aligned}&\{\varphi_0(X_k), \varphi_1(X_k), \cdots, \varphi_{m-1}(X_k)\}\\ &= \left\{X_k, X_k^q, \cdots, X_k^{q^{m-1}}\right\}\\ &= \{X_k, X_{kq}, \cdots, X_{kq^{m-1}}\}.\end{aligned} \tag{6}$$

III. Computing m DFT's Over K With 1 DFT Over F

Given m sequences in $\boldsymbol{K}$

$$\left\{x_n^{(0)}\right\}, \left\{x_n^{(1)}\right\}, \cdots, \left\{x_n^{(m-1)}\right\}, \qquad n = 0, 1, \cdots, N-1,$$

we can compute the DFT of all m sequences simultaneously with one DFT as follows. Choose a basis $\{\beta_0, \beta_1, \cdots, \beta_{m-1}\}$ of $\boldsymbol{F}_{\boldsymbol{K}}$. Form a new sequence $\{x_n\}$ from the m sequences by defining

$$x_n = x_n^{(0)}\beta_0 + x_n^{(1)}\beta_1 + \cdots + x_n^{(m-1)}\beta_{m-1}, \qquad n = 0, 1, \cdots, N-1 \tag{7}$$

Note that the sequences $\{x_n^{(i)}\}$ are in $\boldsymbol{K}$ while the new sequence $\{x_n\}$ is in $\boldsymbol{F}$. From (7), we have that

$$\begin{aligned}\sum_{n=0}^{N-1} x_n W_N^{nk} &= \beta_0 \sum_{n=0}^{N-1} x_n^{(0)} W_N^{nk} + \beta_1 \sum_{n=0}^{N-1} x_n^{(1)} W_N^{nk}\\ &\quad + \cdots + \beta_{m-1} \sum_{n=0}^{N-1} x_n^{(m-1)} W_N^{nk}.\end{aligned}$$

Thus, if we denote the DFT of $\{x_n\}$ by $\{X_n\}$ and the DFT of $\{x_n^{(i)}\}$ by $\{X_n^{(i)}\}$, then the previous equation can be restated as

$$X_k = \beta_0 X_k^{(0)} + \beta_1 X_k^{(1)} + \cdots + \beta_{m-1} X_k^{(m-1)}. \tag{8}$$

The problem is to generate the $X_k^{(l)}$'s from X_k's.

Consider the equivalence class (5) of X_k

$$\begin{aligned}X_k &= \beta_0 X_k^{(0)} + \beta_1 X_k^{(1)} + \cdots + \beta_{m-1} X_k^{(m-1)},\\ X_{kq} &= \beta_0 X_{kq}^{(0)} + \beta_1 X_{kq}^{(1)} + \cdots + \beta_{m-1} X_{kq}^{(m-1)},\\ &\vdots\\ X_{kq^{m-1}} &= \beta_0 X_{kq^{m-1}}^{(0)} + \beta_1 X_{kq^{m-1}}^{(1)} + \cdots + \beta_{m-1} X_{kq^{m-1}}^{(m-1)}.\end{aligned}$$

Since $\{x_n\}$ is an $\boldsymbol{F}$-sequence, the conjugacy relation (4) does not hold for the X_k's. However, the conjugacy relation *does* hold for the $X_k^{(l)}$'s since $\{x_n^{(l)}\}$ are $\boldsymbol{K}$-sequences. Therefore, the above can be rewritten as

$$\begin{aligned}X_k &= \beta_0 X_k^{(0)} + \beta_1 X_k^{(1)} + \cdots + \beta_{m-1} X_k^{(m-1)},\\ X_{kq} &= \beta_0 \left(X_k^{(0)}\right)^q + \beta_1 \left(X_k^{(1)}\right)^q + \cdots + \beta_{m-1} \left(X_k^{(m-1)}\right)^q,\\ &\vdots\\ X_{kq^{m-1}} &= \beta_0 \left(X_k^{(0)}\right)^{q^{m-1}} + \beta_1 \left(X_k^{(1)}\right)^{q^{m-1}} + \cdots\\ &\quad + \beta_{m-1} \left(X_k^{(m-1)}\right)^{q^{m-1}}.\end{aligned}$$

Applying φ_{m-l} to X_{kq^l} and remembering that in a finite field an element raised to the order of field is the element itself, we have

$$\begin{aligned}\varphi_0(X_k) &= \varphi_0(\beta_0)X_k^{(0)} + \varphi_0(\beta_1)X_k^{(1)} + \cdots\\ &\quad + \varphi_0(\beta_{m-1})X_k^{(m-1)}\\ &= \beta_0 X_k^{(0)} + \beta_1 X_k^{(1)} + \cdots + \beta_{m-1} X_k^{(m-1)}\\ &= X_k,\\ \varphi_{m-1}(X_{kq}) &= \varphi_{m-1}(\beta_0)X_k^{(0)} + \varphi_{m-1}(\beta_1)X_k^{(1)} + \cdots\\ &\quad + \varphi_{m-1}(\beta_{m-1})X_k^{(m-1)}\\ &= \beta_0^{q^{m-1}} X_k^{(0)} + \beta_1^{q^{m-1}} X_k^{(1)} + \cdots + \beta_{m-1}^{q^{m-1}} X_k^{(m-1)}\\ &= X_{kq}^{q^{m-1}},\\ &\vdots\\ \varphi_1(X_{kq^{m-1}}) &= \varphi_1(\beta_0)X_k^{(0)} + \varphi_1(\beta_1)X_k^{(1)} + \cdots\\ &\quad + \varphi_1(\beta_{m-1})X_k^{(m-1)}\\ &= \beta_0^q X_k^{(0)} + \beta_1^q X_k^{(1)} + \cdots + \beta_{m-1}^q X_k^{(m-1)}\\ &= X_{kq^{m-1}}^q,\end{aligned}$$

or in matrix notation

$$\begin{pmatrix} X_k \\ X_{kq}^{q^{m-1}} \\ X_{kq^2}^{q^{m-2}} \\ \cdots \\ X_{kq^{m-1}}^{q} \end{pmatrix} = \begin{pmatrix} \beta_0 & \beta_1 & \cdots & \beta_{m-1} \\ \beta_0^{q^{m-1}} & \beta_1^{q^{m-1}} & \cdots & \beta_{m-1}^{q^{m-1}} \\ \beta_0^{q^{m-2}} & \beta_1^{q^{m-2}} & \cdots & \beta_{m-1}^{q^{m-2}} \\ \cdots & \cdots & \cdots & \cdots \\ \beta_0^{q} & \beta_1^{q} & \cdots & \beta_{m-1}^{q} \end{pmatrix} \begin{pmatrix} X_k^{(0)} \\ X_k^{(1)} \\ X_k^{(2)} \\ \cdots \\ X_k^{(m-1)} \end{pmatrix}. \tag{9}$$

From this it is clear that the $X_k^{(l)}$'s are recoverable from the X_k's iff the matrix

$$\begin{aligned}M &\triangleq \begin{pmatrix} \varphi_0(\beta_0) & \varphi_0(\beta_1) & \cdots & \varphi_0(\beta_{m-1}) \\ \varphi_{m-1}(\beta_0) & \varphi_{m-1}(\beta_1) & \cdots & \varphi_{m-1}(\beta_{m-1}) \\ \varphi_{m-2}(\beta_0) & \varphi_{m-2}(\beta_1) & \cdots & \varphi_{m-2}(\beta_{m-1}) \\ \cdots & \cdots & \cdots & \cdots \\ \varphi_1(\beta_0) & \varphi_1(\beta_1) & \cdots & \varphi_1(\beta_{m-1}) \end{pmatrix}\\ &= \begin{pmatrix} \beta_0 & \beta_1 & \cdots & \beta_{m-1} \\ \beta_0^{q^{m-1}} & \beta_1^{q^{m-1}} & \cdots & \beta_{m-1}^{q^{m-1}} \\ \beta_0^{q^{m-2}} & \beta_1^{q^{m-2}} & \cdots & \beta_{m-1}^{q^{m-2}} \\ \cdots & \cdots & \cdots & \cdots \\ \beta_0^{q} & \beta_1^{q} & \cdots & \beta_{m-1}^{q} \end{pmatrix}\end{aligned} \tag{10}$$

is invertible. Fortunately, this is the case.

Theorem 1 [1]: M is invertible iff $\{\beta_0, \beta_1, \cdots, \beta_{m-1}\}$ is a basis of $\boldsymbol{F}_{\boldsymbol{K}}$.

Therefore,

$$\begin{pmatrix} X_k^{(0)} \\ X_k^{(1)} \\ X_k^{(2)} \\ \cdots \\ X_k^{(m-1)} \end{pmatrix} = M^{-1} \begin{pmatrix} \varphi_0(X_k) \\ \varphi_{m-1}(X_{kq}) \\ \varphi_{m-2}(X_{kq^2}) \\ \cdots \\ \varphi_1(X_{kq^{m-1}}) \end{pmatrix} = M^{-1} \begin{pmatrix} X_k \\ X_{kq}^{q^{m-1}} \\ X_{kq^2}^{q^{m-2}} \\ \cdots \\ X_{kq^{m-1}}^{q} \end{pmatrix}. \tag{11}$$

Let us summarize the procedure. Given $\boldsymbol{F}$ and $\boldsymbol{K}$ with $\boldsymbol{K} \subset \boldsymbol{F}$ and $[\boldsymbol{F}:\boldsymbol{K}] = m$, we can compute the DFT of m $\boldsymbol{K}$-sequences $\{x_n^{(l)}\}$ with one $\boldsymbol{F}$-DFT as follows.

1. Choose a basis $\{\beta_0, \beta_1, \cdots, \beta_{m-1}\}$ for $\boldsymbol{F}_{\boldsymbol{K}}$.
2. Form a new sequence $\{x_n\}$ where

$$x_n = x_n^{(0)}\beta_0 + x_n^{(1)}\beta_1 + \cdots + x_n^{(m-1)}\beta_{m-1},$$
$$n = 0, 1, \cdots, N-1.$$

3. Compute the DFT, $\{X_n\}$, of $\{x_n\}$.
4. Compute $\{X_n^{(l)}\}$ from $\{X_n\}$ via (11).

Remark 1: It suffices to solve (11) once per conjugacy class. Since the conjugacy relation (4) holds for $X_n^{(l)}$, any representative member serves to determine the class uniquely.

Remark 2: As is customary in DFT computations, constants of the algorithm (i.e., values that are data-independent such as W_N^k that depends on N and k) are computed "off-line" and are considered cost-free. Thus, the evaluation of M^{-1} is considered inconsequential. Nonetheless, clever choices of basis elements can facilitate the *precomputation* of M^{-1}. We mention two particularly useful bases.

a) *Polynomial Basis:* Let ρ be a primitive element of $\boldsymbol{F}$, then $\{1, \rho, \rho^2, \cdots, \rho^{m-1}\}$ forms a basis for $\boldsymbol{F}_{\boldsymbol{K}}$ called a polynomial basis. With this choice of basis, we have

$$M = \begin{pmatrix} 1 & (\rho) & (\rho)^2 & \cdots & (\rho)^{m-1} \\ 1 & \left(\rho^{q^{m-1}}\right) & \left(\rho^{q^{m-1}}\right)^2 & \cdots & \left(\rho^{q^{m-1}}\right)^{m-1} \\ 1 & \left(\rho^{q^{m-2}}\right) & \left(\rho^{q^{m-2}}\right)^2 & \cdots & \left(\rho^{q^{m-2}}\right)^{m-1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & (\rho^q) & (\rho^q)^2 & \cdots & (\rho^q)^{m-1} \end{pmatrix}. \tag{12}$$

With this choice, M is a Vandermonde matrix; therefore its inverse can be computed in $O(N^2)$ time (as opposed to $O(N^3)$ time for a general matrix) by well-known methods.

b) *Normal Basis:* If we take the basis to be a normal basis, i.e., a basis of the form $\left\{\alpha, \alpha^q, \alpha^{q^2}, \cdots, \alpha^{q^{m-1}}\right\}$, then

$$M = \begin{pmatrix} \alpha & \alpha^q & \alpha^{q^2} & \cdots & \alpha^{q^{m-1}} \\ \alpha^{q^{m-1}} & \alpha & \alpha^q & \cdots & \alpha^{q^{m-2}} \\ \alpha^{q^{m-2}} & \alpha^{q^{m-1}} & \alpha & \cdots & \alpha^{q^{m-3}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \alpha^q & \alpha^{q^2} & \alpha^{q^3} & \cdots & \alpha \end{pmatrix}. \tag{13}$$

M is seen to be a circulant matrix and thus has a circulant inverse. We can in fact get M^{-1} without any calculation in this case. Let $\left\{\eta, \eta^q, \eta^{q^2}, \cdots, \eta^{q^{m-1}}\right\}$ be the (unique) *dual basis* of $\left\{\alpha, \alpha^q, \alpha^{q^2}, \cdots, \alpha^{q^{m-1}}\right\}$

Claim:

$$M^{-1} = \begin{pmatrix} \eta & \eta^{q^{m-1}} & \eta^{q^{m-2}} & \cdots & \eta^q \\ \eta^q & \eta & \eta^{q^{m-1}} & \cdots & \eta^{q^2} \\ \eta^{q^2} & \eta^q & \eta & \cdots & \eta^{q^3} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \eta^{q^{m-1}} & \eta^{q^{m-2}} & \eta^{q^{m-3}} & \cdots & \eta \end{pmatrix}.$$

Proof:

$$\left(MM^{-1}\right)_{ij} = \sum_{k=0}^{m-1} \alpha^{q^{m-i+k}} \eta^{q^{m-j+k}}$$
$$= \operatorname{tr}\left(\alpha\eta^{q^{i-j}}\right) = \delta_{ij}. \qquad \square$$

Tables of finite fields and normal bases can be found in [1], [7], [8].

IV. The Real/Complex Case Revisited

Though the previous procedure is developed in the context of finite fields, the derivation is valid, mutatis mutandis, for the case $\boldsymbol{K} = \boldsymbol{R}$ and $\boldsymbol{F} = \boldsymbol{C}$. By making use of the automorphism group $\operatorname{Aut}_{\boldsymbol{K}}\boldsymbol{F}$, we have ensured the validity of the derivation in all respects except for the statement of the conjugacy relationship. By replacing all expressions of conjugacy relations in finite fields by their counterparts in the complex field, we give an algorithm for computing two real DFT's with one complex DFT. It will be seen that the algorithm includes, as a special case, the usual method for performing this computation.

For $\boldsymbol{K} = \boldsymbol{R}$ and $\boldsymbol{F} = \boldsymbol{C}$, we have $[\boldsymbol{C}:\boldsymbol{R}] = 2$ and $\operatorname{Aut}_{\boldsymbol{R}}\boldsymbol{C} = \{\varphi_0, \varphi_1\}$, where

$$\varphi_0 : a + ib \mapsto a + ib,$$
$$\varphi_1 : a + ib \mapsto a - ib.$$

Let $\{\beta_0, \beta_1\}$ be a basis for $\boldsymbol{C}_{\boldsymbol{R}}$. Given 2 real sequences $\{x_n^{(0)}\}$ and $\{x_n^{(1)}\}$, form the complex sequence $\{x_n\}$ by defining

$$x_n = x_n^{(0)}\beta_0 + x_n^{(1)}\beta_1, \qquad n = 0, 1, \cdots, N-1.$$

Denote the DFT of $\{x_n^{(0)}\}$, $\{x_n^{(1)}\}$, and $\{x_n\}$ by $\{X_n^{(0)}\}$, $\{X_n^{(1)}\}$, and $\{X_n\}$, respectively. Then the three DFT's are related by (11) as

$$\begin{pmatrix} X_k^{(0)} \\ X_k^{(1)} \end{pmatrix} = M^{-1} \begin{pmatrix} \varphi_0(X_k) \\ \varphi_1(X_{-k}) \end{pmatrix},$$

where (see (10))

$$M = \begin{pmatrix} \varphi_0(\beta_0) & \varphi_0(\beta_1) \\ \varphi_1(\beta_0) & \varphi_1(\beta_1) \end{pmatrix}.$$

In other words,

$$\begin{pmatrix} X_k^{(0)} \\ X_k^{(1)} \end{pmatrix} = \frac{1}{\beta_0\beta_1^* - \beta_0^*\beta_1} \begin{pmatrix} \beta_1^* & -\beta_1 \\ -\beta_0^* & \beta_0 \end{pmatrix} \begin{pmatrix} X_k \\ X_{-k}^* \end{pmatrix}. \tag{14}$$

In particular, for $\beta_0 = 1$ and $\beta_1 = i$, we have

$$\begin{pmatrix} X_k^{(0)} \\ X_k^{(1)} \end{pmatrix} = \frac{1}{-2i} \begin{pmatrix} -i & -i \\ -1 & 1 \end{pmatrix} \begin{pmatrix} X_k \\ X_{-k}^* \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ -i & i \end{pmatrix} \begin{pmatrix} X_k \\ X_{-k}^* \end{pmatrix}, \tag{15}$$

which is the usual method of computing two real DFT's with one complex DFT [6].

V. Conclusion

For finite fields $\boldsymbol{F}$ and $\boldsymbol{K}$ with $\boldsymbol{K} \subset \boldsymbol{F}$ and $[\boldsymbol{F}:\boldsymbol{K}] = m$ we have shown that a single DFT over $\boldsymbol{F}$ has complexity approximately m times that of DFT over $\boldsymbol{K}$ by showing how to compute m $\boldsymbol{K}$-DFT's simultaneously with one $\boldsymbol{F}$-DFT. The result can be generalized to arbitrary fields $\boldsymbol{F}$ and $\boldsymbol{K}$ by assuming additionally the normality and separability of $\boldsymbol{F}$ over $\boldsymbol{K}$. Thus, the precise condition required for this argument to work for arbitrary fields $\boldsymbol{F}$ and $\boldsymbol{K}$ is that $\boldsymbol{F}$ is a *finite Galois* extension of $\boldsymbol{K}$. In the case considered here, since all finite extensions of finite fields are Galois, the only assumption needed is that $[\boldsymbol{F}:\boldsymbol{K}]$ is finite. For details, see [9].

REFERENCES

[1] R. Lidl and H. Niederreiter, Finite Fields, *Encyclopedia of Mathematics and Its Applications,* vol. 20. Reading, MA: Addison-Wesley, 1983.
[2] L. C. Grove, *Algebra.* Orlando, FL: Academic Press, 1983.
[3] T. W. Hungerford, *Algebra.* New York: Springer-Verlag, 1974.
[4] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error Correcting Codes.* Amsterdam: North-Holland, 1977.
[5] R. E. Blahut, *Theory and Practice of Error Control Codes.* Reading, MA: Addison-Wesley, 1983.
[6] ——, *Fast Algorithms for Digital Signal Processing.* Reading, MA: Addison-Wesley, 1986.
[7] J. H. Conway, *A Tabulation of Some Information Concerning Finite Fields, Computers in Mathematical Research.* Amsterdam: North-Holland, 1968.
[8] J. D. Alanen and D. E. Knuth, *Tables of Finite Fields.* Sankhya, Series A, vol. 26, 1964, pp. 305–328.
[9] J. Hong, "Finite field transforms for signal processing," PhD thesis, in preparation, Columbia Univ., 1992.

Random Interactions in Higher Order Neural Networks

Pierre Baldi and Santosh S. Venkatesh

***Abstract*—Recurrent networks of polynomial threshold elements with random symmetric interactions are studied. Precise asymptotic estimates are derived for the expected number of fixed points as a function of the margin of stability. In particular, it is shown that there is a critical range of margins of stability (depending on the degree of polynomial interaction) such that the expected number of fixed points with margins below the critical range grows exponentially with the number of nodes in the network, while the expected number of fixed points with margins above the critical range decreases exponentially with the number of nodes in the network. The random energy model is also briefly examined and links with higher order neural networks and higher order spin glass models made explicit.**

***Index Terms*— Neural networks, spin glasses, polynomial threshold elements, fixed points, Laplace's method.**

I. INTRODUCTION

Recurrent networks of formal neurons have been popular in a variety of computational applications. The model neurons in such structures are typically linear threshold elements which compute the sign of a linear form of the inputs. A recurrent network results when such elements are fully interconnected, and as in any recurrent system, the fixed points are important in the characterization of the computations done by the structure. A particular case of interest results when the interconnections between neurons are symmetric: in such cases network dynamics are regulated by a Hamiltonian or energy function (cf. Hopfield [1], for instance). In such an instance, we can imagine the state space of the network to be embedded in an energy landscape with fixed points residing at energy minima. A classical application of such networks is in associative memory where neural interactions are adjusted so that memories are stored as local attractors.

We consider here a natural extension of the model to recurrent networks comprised of higher order neurons that compute the sign of a *polynomial* form of the inputs. The added degrees of freedom in specifying the polynomial interaction coefficients can be expected to enrich the computational dynamics that result. Distinct features emerge, however, in the analysis of these structures depending on whether the higher order interactions are programmed (or "learned") or random.

In the programmed scenario, the goal is to tailor the higher order interaction coefficients so as to obtain desired dynamical behaviors; this leads naturally to questions of capacity and efficiency of higher order networks of given degree of polynomial interaction. In two concurrent papers [2], [3], we present rigorous results on algorithmic capacity and efficiency in programmed situations for higher order networks (cf. also Newman [4]). The main results can be summarized briefly as follows: the computational gains in higher order networks parallel the extra degrees of freedom in specifying the polynomial interaction coefficients; in particular, regardless of the algorithm used to specify the interaction coefficients, the information storage capability of a higher order network is of the order of one bit per interaction coefficient.

Higher order systems where the polynomial interactions are random may be useful as models of disordered systems in statistical physics (spin glasses), or of neural networks, before any learning has occurred, or in the limiting case when too much learning has occurred (the onset of senility!). These will be our focus of analysis in this paper: in particular, we consider recurrent, higher order neural networks with symmetric, random polynomial interactions. We characterize the fixed points of these structures according to their *margin of stability*[1] that is a measure of how stable a fixed point is with respect to perturbations. Our main result may be informally stated as follows:

> There exists a critical range of margins of stability (depending on the degree of polynomial interaction) such that the expected number of fixed points with margins of stability below the critical range increases exponentially in the size of the network while the expected number of fixed points with margins of stability above the critical range decays exponentially as the size of the network is increased.

There is thus a threshold phenomenon in evidence for the expected number of fixed points around the critical range of the margin of stability. The fact that for a certain range of margins the expected number of fixed points grows exponentially with the number of nodes in the network is not unexpected; more counter-intuitive, perhaps, is the existence of a critical margin of stability above which the expected number of fixed points actually decays as more nodes are added. We also provide exact asymptotic expressions for the coefficients and exponents in the regime of exponential behavior, and evaluate the critical margins of stability. While considerable attention has been focused on spin glass models in the statistical physics literature, at

[1] In this context, this notion is due to Komlós and Paturi [9].

Manuscript received October 3, 1988; revised May 2, 1991. P. Baldi was supported by NSF Grant DMS-8800322. S. S. Venkatesh was supported in part by NSF Grant EET-8709198 and in part by the Air Force Office of Scientific Research under Grant AFOSR 89–0523. This work was presented in part at the Sixth International Conference on Mathematical Modeling, St. Louis, MO, 1987; and in part at the Conference on Neural Information Processing Systems, Denver, CO, 1987.

P. Baldi is with the Jet Propulsion Laboratories, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109. He is also with the Division of Biology, California Institute of Technology, Pasadena, CA 91125.

S. S. Venkatesh is with the Department of Electrical Engineering, University of Pennsylvania, Philadelphia, PA 19104.

IEEE Log Number 9203003.