

P20

GEORGIA INSTITUTE OF TECHNOLOGY  
OFFICE OF CONTRACT ADMINISTRATION  
SPONSORED PROJECT INITIATION

Date: 2/10/79

Project Title: Representation and Processing of Signals in Two-Dimensional Form

Project No: E-21-656

*Green card*

CO-Project Directors: Dr. R. M. Mersereau and Dr. R. W. Schafer

Sponsor: National Science Foundation

Agreement Period: From 1/1/79 Until 9/30/82  
~~6/30/80~~ (Grant Period)

Type Agreement: Grant No. ENG-7817201

Amount: \$35,868 NSF  
10,767 GIT (E-21-341)  
\$46,635 TOTAL

Reports Required: Annual Progress Report (s) (if grant is extended); Final Project Report

Sponsor Contact Person (s):

Technical Matters

(NSF Program Official)  
Elias Schutzman  
Program Director  
Electrical and Optical Communications  
Program  
Electrical Sciences and Analysis Section  
Division of Engineering  
National Science Foundation  
Washington, D.C. 20550  
202-632-5881

Contractual Matters

(thru OCA)

(NSF Grants Official)  
Ms. Mary Frances O'Connell  
Grants Specialist, Area 4  
MPE/BBS/SE Branch  
Division of Grants and Contracts  
National Science Foundation  
Washington, D.C. 20550  
202-632-2858

Defense Priority Rating: n/a

Assigned to: Electrical Engineering (School/Laboratory)

COPIES TO:

Project Director  
Division Chief (EES)  
School/Laboratory Director  
Dean/Director-EES  
Accounting Office  
Procurement Office  
Security Coordinator (OCA)  
Reports Coordinator (OCA)

Library, Technical Reports Section  
EES Information Office  
EES Reports & Procedures  
Project File (OCA)  
Project Code (GTRI)  
Other \_\_\_\_\_

SPONSORED PROJECT TERMINATION SHEET

5/4/83  
M-380

Date 5/4/83

Project Title: Representation and Processing of Signals in Two-Dimensional Form

Project No: E-21-656

Project Director: Dr. R. M. Mersereau & Dr. R. W. Schafer

Sponsor: National Science Foundation

Effective Termination Date: 9/30/82

Clearance of Accounting Charges: 9/30/82

Grant/Contract Closeout Actions Remaining:

- Final Invoice and Closing Documents
- Final Fiscal Report Acctg. (FCTR)
- Final Report of Inventions (only if positive)
- Govt. Property Inventory & Related Certificate
- Classified Material Certificate
- Other \_\_\_\_\_

Assigned to: Elect. Engr. (School/Laboratory)

COPIES TO:

Administrative Coordinator  
 Research Property Management  
 Accounting  
 Procurement/EES Supply Services

Research Security Services  
Reports Coordinator (OCA)  
 Legal Services (OCA)  
 Library

EES Public Relations (2)  
 Computer Input  
 Project File  
 Other Mersereau/Schafer

PLEASE READ INSTRUCTIONS ON REVERSE BEFORE COMPLETING

PART I-PROJECT IDENTIFICATION INFORMATION

1. Institution and Address Georgia Tech Research Institute Georgia Institute of Technology Atlanta, Georgia 30332	2. NSF Program Elec.&Optical Communication	3. NSF Award Number ECS78-17201
	4. Award Period From 1/79 To 9/82	5. Cumulative Award Amount \$122,251
6. Project Title Representation and Processing of Signals in Two-Dimensional Form		

PART II-SUMMARY OF COMPLETED PROJECT (FOR PUBLIC USE)

This research was concerned with finding and exploiting efficient and novel two-dimensional representations of both one-dimensional signals, such as speech and two-dimensional signals, such as photographic images. As such, the research was divided into two parts. A 2-D time-frequency representation for speech was extended and used to enhance the nearly unintelligible speech which results when a submerged diver is breathing a helium rich atmosphere under pressure. Using acoustic models for speech production, a non-linear procedure was developed to alter the frequency spectrum of the distorted speech and a two-dimensional noise stripping procedure was developed to improve the signal-to-noise ratio of the restored speech. These algorithms both improved the quality of the distorted speech, but the noise-stripping procedure did not improve its intelligibility. The second part of this research, which was concerned with efficient representations of two-dimensional signals, considered signals which were sampled on arbitrary periodic sampling lattices. These representations are known to be efficient, but it was not known how signal processing algorithms could be performed using data stored in this format. The current research showed essentially that any signal processing operations that could be performed on one-dimensional signals could be performed on any of these multidimensional representations. These resulting algorithms generally required less computation than when traditional sampling strategies were used. These results are directly applicable not only to problems in multidimensional signal processing, but also to the design of phased-array antennas and beamformers.

PART III-TECHNICAL INFORMATION (FOR PROGRAM MANAGEMENT USES)

1. ITEM (Check appropriate blocks)	NONE	ATTACHED	PREVIOUSLY FURNISHED	TO BE FURNISHED SEPARATELY TO PROGRAM	
				Check (✓)	Approx. Date
a. Abstracts of Theses		X			
b. Publication Citations		X			
c. Data on Scientific Collaborators		X			
d. Information on Inventions		X			
e. Technical Description of Project and Results				X	5/15/83
f. Other (specify) copies of publications		X			
2. Principal Investigator/Project Director Name (Typed) Russell M. Mersereau Ronald W. Schafer	3. Principal Investigator/Project Director Signature			4. Date 4/20/83	



GEORGIA INSTITUTE OF TECHNOLOGY  
SCHOOL OF ELECTRICAL ENGINEERING  
ATLANTA, GEORGIA 30332

TELEPHONE: (404) 894-2961

July 29, 1983

Post Award Projects Branch  
Division of Grants and Contracts  
National Science Foundation  
1800 G St., N. W.  
Washington, DC 20550

SUBJECT: Final Report (Technical Summary of Research)  
Project Directors: Dr. R. M. Mersereau  
Dr. R. W. Schafer  
Grant No. ECS-7817201  
"Representation and Processing of Signals in  
Two-Dimensional Form"  
Period Covered: 1/1/79 - 9/30/82

Dear Sirs:

The subject Technical Summary of Research is forwarded in conformance with the contract/grant specifications.

Should you have any questions or comments regarding this report, please contact the project director or the undersigned.

Sincerely,

Marsha Segraves U  
Administrative Asst.

/ms  
Distribution:  
Addressee, 2 copies  
R. M. Mersereau  
R. W. Schafer  
File (E21-656)

**TECHNICAL SUMMARY OF RESEARCH**

**ON**

**NSF GRANT NO. ECS-7817201**

Representation and Processing

of

Signals in Two-Dimensional Form

By

R. M. Mersereau

R. W. Schafer

Georgia Institute of Technology  
School of Electrical Engineering  
Atlanta, Georgia 30332

This technical summary is Appendix E of the Final Project Report on this grant which was submitted with Form 98A on April 20, 1983.

## EFFICIENT REPRESENTATIONS OF MULTIDIMENSIONAL SIGNALS

There are many generalizations of one-dimensional (1-D) periodic sampling which permit the exact representation of a bandlimited continuous function of several independent variables. By far the most common of these is rectangular sampling, which corresponds to evaluating the function on sampling locations that form a regular hypercubic lattice. It was shown in [1] that one form of nonrectangular sampling, hexagonal sampling and its higher dimensional generalizations, resulted in a lower sampling density and more efficient signal processing algorithms than traditional rectangular sampling for isotropically bandlimited signals.

There are several applications of multidimensional digital signal processing for which nonrectangular sampling may be more suited to a problem than rectangular sampling. Phased array antennas, for example, are typically designed with a hexagonal arrangement of elements. This allows fewer elements to be used and also provides for a more symmetric response. Measurements made in the near field of such an array are more accurate and more easily interpreted if hexagonal sampling is used. Similar results might be expected in measuring fields in or near crystals and in processing the results. Here the sampling lattice should be related to the crystal lattice. Serra [2] argues that image processing should be performed on hexagonal grids, particularly when morphological ideas such as connectivity are important.

The purpose of this research has been to show that most common signal processing operations, such as the implementation and design of both recursive and nonrecursive digital filters, multi-dimensional decimation and interpolation, Fourier analysis, and discrete Fourier transform computation can be performed on a signal defined on any multidimensional periodic lattice, to derive the resulting algorithms, and to analyze their computational

efficiency. Our results have included not only the attainment of all of these goals, but also a clearer understanding of multidimensional digital signal processing in the rectangular case.

#### A. Linear, Shift-Invariant Systems for Signals Defined on Arbitrary Lattices

A D-dimensional lattice can be defined as the set of integer linear combinations of D linearly independent vectors. The points in a lattice are thus defined by

$$\underline{r} = \underline{V} \underline{n}$$

where  $\underline{n}$  is an integer (column) vector and  $\underline{V}$  is a (nonunique)  $D \times D$  matrix associated with the lattice known as the sampling matrix.

Let us consider a D-dimensional, linear, shift-invariant system whose input is a complex sinusoid of the form

$$x(\underline{n}) = \exp[j\omega^T \underline{V} \underline{n}].$$

The output of this system is given by

$$y(\underline{n}) = H(\omega) \exp[j\omega^T \underline{V} \underline{n}]$$

where

$$H(\omega) = \sum_{\underline{n}=-\infty}^{\infty} h(\underline{n}) \exp[-j\omega^T \underline{V} \underline{n}]$$

is known as the frequency response of the system. The impulse response,  $h(\underline{n})$ , is the response of the system to the input

$$\delta(\underline{n}) = \begin{cases} 1 & \underline{n} = \underline{0} \\ 0 & \underline{n} \neq \underline{0} \end{cases}$$

As part of this research we looked at a variety of techniques for designing both nonrecursive (FIR) and recursive (IIR) digital filters to approximate an arbitrary ideal frequency response for the special case of 2-D hexagonal filters. These results are summarized in [3] for FIR filters. The three most common methods for the design of rectangular FIR filters are through the use of windows [4], through equiripple design algorithms [5], and through the use of McClellan transformations [6]. All three of these techniques can be generalized and in [3] rectangular and hexagonal versions of these algorithms were compared. The result of that study indicated that hexagonal lowpass FIR filters designed by any of these methods could be implemented with half the computational effort of their rectangular counterparts. We also showed that it was possible to design hexagonal IIR filters and control their stability [7,8]. The approach used here was to observe that the impulse response of a hexagonal filter could be linearly mapped onto a rectangular lattice with support on one quadrant. The frequency responses of the two filters were also related by a linear transformation. Thus a hexagonal design can be performed by approximating a prewarped ideal frequency response with a rectangular filter and inverse transforming the resulting design. Since stability is preserved under this transformation, this mapping provides a means for studying the stability of a recursive system defined on any periodic lattice.

## B. Decimation and Interpolation of Signals

An important question in dealing with signals on an arbitrary lattice is the conversion of such a representation to one on a different lattice. Such a problem must be addressed, for example, when a hexagonally sampled signal must be displayed on a rectangular raster graphics display system. Two solutions to this problem were developed, one of which is approximate and one of which can be made arbitrarily close to exact.

The approximate technique [9] makes use of the generality of the DFT in the multi-dimensional case. Discrete Fourier transforms can be defined which relate a spatial signal defined on one lattice to samples of its Fourier transform defined on a different lattice [10,12]. One can thus compute the DFT of a signal evaluated with respect to one sampling lattice and then perform an inverse Fourier transform with respect to another. The approach is inexact because of inevitable spatial aliasing.

The exact approach [12] mimics the one-dimensional solution to the decimation (reduction of the number of samples) and interpolation (increase in the number of samples) problems. It finds an intermediate sparse lattice for which both the initial and destination lattices are sublattices. The original signal is mapped onto the sparse lattice, filtered, and downsampled onto the destination lattice. The derivation of this algorithm makes straightforward use of the sampling theorem for multidimensional lattices.

## C. Evaluation of Discrete Fourier Transforms On Arbitrary Periodic Lattices

The major portion of our research was concerned with the efficient evaluation of the spectra of signals which are defined on an arbitrary lattice. It was shown [10-12] that samples of the Fourier transform of such a signal could be evaluated on any periodic lattice using a generalization of the normal rectangular multidimensional discrete Fourier transform. Attention was

then directed toward finding efficient algorithms for evaluating this general DFT when the number of points in the transform was a composite number.

In the one-dimensional case the three common algorithms for evaluating discrete Fourier transforms are the Cooley-Tukey FFT algorithm, the prime-factor algorithm, and the Winograd Fourier transform. All three of these algorithms have been generalized to the case where the DFTs are defined over lattices. The Cooley-Tukey algorithm was discussed in [11-14] and the prime-factor algorithm was discussed in [15,16]. Our results on the Winograd generalization are being prepared for publication. The details of all of these algorithms are fairly abstract and will not be discussed here except to observe that they depend upon the factorization of a small  $D \times D$  integer matrix, just as in the one-dimensional case, all of the efficient algorithms depend on the length of the transform being factorable. When this matrix can be factored, the evaluation of a large multi-dimensional DFT can be performed by evaluating a number of smaller DFTs with a resulting saving of computation. Another approach for evaluating an arbitrary DFT relies on putting this critical integer matrix in Smith normal form. The computation of the DFT then reduces to scrambling the data, evaluating a rectangular DFT, and unscrambling the result.

Additional effort has been directed to programming of these algorithms.

#### D. Other Work

In a related study [4] we looked at the window method for the design of FIR filters. The purpose of this study was to compare alternate window formulations and to develop guidelines which would relate the filter order to approximation errors. Rectangular and circular regions of support were found to perform similarly for lowpass designs.

Another study [18] looked at quality measures for coded photographic imagery. Many coding algorithms are designed to minimize mean-squared error, but this is known to correlate poorly with subjective judgements of image quality. On the other hand, subjective tests themselves are expensive, time-consuming, and difficult to quantify. The goal of this study was the development of objective quality measures which would correlate well with subjective performance. Various nonlinear functions of the signal-to-noise ratio and the signal-to-noise ratios in disjoint frequency bands were found to work reasonably well. The correlation coefficient between subjective and objective measure was .9.

## TWO-DIMENSIONAL PROCESSING OF ONE-DIMENSIONAL SIGNALS

In this part of the research, we investigated the use of two-dimensional representations of intrinsically one-dimensional signals such as time waveforms. In particular we investigated time-frequency, or short-time Fourier, representations of the form

$$X(n,k) = \sum_{m=-\infty}^{\infty} x(m)w(n-m)e^{-j\omega_k n}$$

where  $x(m)$  is the one-dimensional signal,  $w(n-m)$  is a sliding "window" function,  $n$  is an index proportional to time, and  $\omega_k = 2\pi k/N$ . Thus, the other index,  $k$ , is proportional to frequency. Theoretically, it is well known that  $X(n,k)$  is an exact representation since we can recover  $x(n)$  from an equation of the form

$$x(n) = \frac{1}{Nw(0)} \sum_{k=0}^{N-1} X(n,k)e^{j\omega_k n}$$

Such representations have been used extensively in speech processing and quite a bit is known about the properties of time-frequency representations for certain kinds of signals.

Our research in this area had as its major goal, the study of two-dimensional processing techniques which could be applied to  $X(n,k)$ . That is we were concerned with transformations of the form

$$Y(n,k) = H[X(n,k)]$$

where  $H[\cdot]$  represents some two-dimensional computational algorithm. In order to study this question it was necessary to examine the properties of  $X(n,k)$  for an interesting class of signals. Thus, we chose to investigate problems related to "helium speech;" i.e., speech produced by deep-sea divers breathing a mixture of helium and oxygen at elevated pressure. Since speech produced in this environment is virtually unintelligible due to radical shifts of vocal tract resonances as well as noisy transmission conditions, a major problem is the restoration of the speech to a more natural quality. It turns out that 2D processing of the time-frequency representation is ideally suited for this problem. The results of our research in this area are summarized below.

#### E. Design of Filter Banks for Short-Time Fourier Analysis

In time-frequency analysis/synthesis, it is essential to achieve efficient computational implementations. A significant result of our research is an algorithm for designing recursive filters for decimation and interpolation of the two-dimensional representation,  $X(n,k)$  [19]. Since the filters operate only in the time dimension, they can be used in any problem where decimation or interpolation is required.

#### F. Acoustic Theory of Helium Speech Production

In order to apply the processing techniques described above, it is necessary to have an accurate model for the signal. In the helium speech case, the basic frequency-shift effects are well known. However, a detailed physical model for the unusual acoustic wave phenomena was not readily available. Thus an important aspect of the research was the development of this model and the use of this model to determine the properties of the short-time Fourier transform of helium speech [20]. The research in this area sheds some new light on empirically observed properties of helium speech.

### G. Development of Two-Dimensional Processing Algorithms for Helium Speech

The major contribution of this research was the development of adaptive two-dimensional processing algorithms for restoring speech intelligibility and removing background noise. The details of these algorithms are given in references [20-22]. The algorithms were evaluated using standard intelligibility test procedures and were found to improve intelligibility significantly [20,22].

It is interesting to note that the algorithms which were developed in this research were adopted by a Norwegian group and were implemented in real-time on a fast array processor [23]. Their report suggests that our system is superior in performance to presently available systems. It is also worth noting that with presently available high-speed microcomputer chips, the system could be implemented very economically. A final point is that the techniques developed in this research may also be applicable in creating highly sophisticated hearing aids for the hearing-impaired, since spectral shifting and noise reduction are essential operations in utilizing residual hearing of deaf persons [24].

## REFERENCES

References with an asterisk (\*) in the left margin were supported by this grant.

- [1] R. M. Mersereau, "The processing of hexagonally sampled two-dimensional signals," Proc. IEEE, vol. 67, pp. 930-949, May 1979.
- [2] J. Serra, "Image Analysis and Mathematical Morphology", Academic Press, 1982.
- \* [3] R. M. Mersereau, T. H. Joo, and T. C. Speake, "A comparison of hexagonally and rectangularly sampled two-dimensional FIR filters," 1980 International Conference on Acoustics, Speech and Signal Processing, pp. 729-732.
- \* [4] T. C. Speake and R. M. Mersereau, "A note on the use of windows for two-dimensional FIR filter design," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-29, No. 1, Feb. 1981.
- [5] D. B. Harris and R. M. Mersereau, "A comparison of algorithms for minimax design of two-dimensional linear phase FIR digital filters," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-25, Dec. 1977.
- [6] R. M. Mersereau, W. F. G. Mecklenbrauker, and T. F. Quatieri, "McClellan transformations for two-dimensional digital filtering: I-Design," IEEE Trans. Circuits and Systems, vol. CAS-23, pp. 405-414, July 1976.
- [7] G. A. Shaw and R. M. Mersereau, "Design of two-dimensional recursive filters with arbitrary support using quarter-plane design algorithms," Proc. IEEE, vol. 67, No. 7, July 1979.
- \* [8] P. A. Ramamoorthy, "Two-dimensional hexagonal digital recursive filters," 1981 Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 712-715.
- \* [9] T. C. Speake and R. M. Mersereau, "An interpolation technique for periodically sampled two-dimensional signals," 1981 Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 1010-1013.
- \* [10] R. M. Mersereau and T. C. Speake, "Multidimensional digital signal processing from arbitrary periodic sampling rasters," Int. Conf. on Digital Signal Processing, pp. 93-101, Sept. 1981. Also presented at NSF Joint USA-Italy Workshop on DSP, Portovenere, Italy, Aug. 1981.
- \* [11] R. M. Mersereau and T. C. Speake, "Generalized Cooley-Tukey algorithms for evaluation of multidimensional discrete Fourier transforms," Fourth Int. Conf. on the Analysis and Optimization of Systems, pp. 744-762, Dec. 1982.

- \* [12] R. M. Mersereau and T. C. Speake, "The processing of periodically sampled multidimensional signals," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-31, pp. 188-194, Feb. 1983.
- \* [13] R. M. Mersereau and T. C. Speake, "A unified treatment of Cooley-Tukey algorithms for the evaluation of the multidimensional DFT," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-29, pp. 1011-1018, Oct. 1981.
- \* [14] T. C. Speake and R. M. Mersereau, "Evaluation of two-dimensional discrete Fourier transforms via generalized FFT algorithms," 1981 Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 1006-1009.
- \* [15] R. M. Mersereau, E. W. Brown, and A. Guessoum, "Evaluation of multidimensional DFTs on arbitrary sampling lattices," Sixteenth Asilomar Conference on Circuits, Systems, and Computers, pp. 300-302, 1982.
- \* [16] R. M. Mersereau, E. W. Brown, and A. Guessoum, "Row-Column algorithms for the evaluation of multidimensional DFTs on arbitrary periodic sampling lattices," 1983 Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 1264-1267.
- \* [18] B. Girod, "Objective quality measures for the design of digital image transmission systems," 1981 Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 1132-1135.
- \* [19] M. A. Richards, "Application of Deczky's program for recursive filter design to the design of recursive decimators," Vol. ASSP-30, No. 5, Oct. 1982 pp. 811-814.
- \* [20] M. A. Richards, "Helium speech enhancement using the short-time Fourier transform," Ph.D. Thesis, Georgia Institute of Technology, Atlanta, February 1982.
- \* [21] M. A. Richards, "A system for helium speech enhancement using the short-time Fourier transform," Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, April 1981, pp. 1097-1100.
- \* [22] M. A. Richards, "Helium speech enhancement using the short-time Fourier transform," IEEE Trans. Acoustics, Speech, and Signal Processing, Vol. ASSP-30, No. 6, Dec. 1982, pp. 841-853.
- \* [23] M. Vestrheim, S. Hatlestad, E. Belcher, and K. Slethei, "Deep ex-81 diver communications," Norwegian Underwater Technology Center, Rep. 17-82, Jan 1982.
- [24] R. W. Schafer, "Speech processing for the hearing impaired," The Vanderbilt Hearing-Aid Report, Ed. by G. A. Studebaker and F. H. Bess., 1982.

## APPENDIX A

### ABSTRACTS OF THESES COMPLETED

Two doctoral students were supported by this research grant, Mark A. Richards and Theresa C. Speake. Dr. Richards' thesis was completed in February 1982 and was titled "Helium Speech Enhancement Using the Short-Time Fourier Transform." A summary of that thesis appears in this Appendix. Ms. Speake completed all of the work for her thesis but left Georgia Tech before actually writing up the dissertation. Her thesis was to have been titled "Generalized Sampling and Digital Processing of Multi-Dimensional Bandlimited Signals." The primary results of that research are described in many of the publications which are associated with this grant.

In addition to these two doctoral students, one Masters student was partially supported under this grant. The abstract of Mr. Bernd Girod's M.S. thesis entitled "Objective Quality Measures for the Design of Digital Image Transmission Systems" is also included in this appendix.

HELIUM SPEECH ENHANCEMENT USING THE  
SHORT-TIME FOURIER TRANSFORM

A THESIS

Presented to

The Faculty of the Division of Graduate Studies

By

Mark Andrew Richards

In Partial Fulfillment  
of the Requirements for the Degree  
Doctor of Philosophy  
in the School of Electrical Engineering

Georgia Institute of Technology

February, 1982

## SUMMARY

Breathing air at high pressure leads to severe physiological hazards. In particular, the nitrogen present in air can cause nitrogen narcosis and the "bends." Persons who must operate in hyperbaric environments, such as deep sea divers, avoid these problems by breathing an artificial atmosphere, usually composed of helium and oxygen ("heliox"). However, the acoustic properties of hyperbaric heliox differ drastically from those of air, with the result that speech uttered in a heliox atmosphere is virtually unintelligible. The purpose of this thesis is to develop, simulate, and evaluate a new system for enhancing helium speech so as to improve its intelligibility.

The acoustic tube theory of speech is reviewed to indicate how important speech features such as formant frequencies, bandwidths, and amplitudes depend on atmospheric properties. This analysis leads to predictions for the relationship between these speech features in helium speech and normal speech. Two significant points are raised concerning the modeling of helium speech phenomena. The first is a prediction of an increase in formant bandwidths in helium speech, contradicting some previous claims. The second is a discussion of the origins of reported losses in upper formant amplitudes.

The system to be proposed is based on the use of a short-time Fourier transform (STFT) representation of the speech signal. Accordingly, the basic theory of signal processing via the STFT is

reviewed. The choice of sampling rates for the STFT is discussed extensively. Although the basic equations governing this choice are not new, no thorough discussion of their implications has been given. The terminal analog model of speech is used to derive a model for the STFT of speech. This model, which is the discrete equivalent to a recently developed model for the continuous-time STFT, is more general than the traditional quasistationary model in that it allows some variation of the local pitch and vocal tract configuration. However, the simpler quasistationary model is found adequate for this application.

The acoustic analysis and empirical evidence are applied to the speech STFT model to derive an explicit relationship between the STFTs of helium and normal speech, thereby implying a simple enhancement algorithm. The key step in the algorithm is the estimation of the envelope of the STFT in each frame. A simple new algorithm, the "piecewise-linear method", is devised and is shown to be superior to three other envelope estimation techniques in this application. Also, a "spectral subtraction" noise reduction technique is incorporated.

Major computational issues are also discussed. A simple method for early bandwidth reduction is described.

The system is evaluated using formal intelligibility tests. It is found that, in its best configuration, the proposed system operates on a par with the best currently available commercial device. There are some indications that the proposed system may have better performance, but this is not conclusively established. The noise reduction process is found to be detrimental to the intelligibility of the

enhanced speech. Also, the importance of a nonlinear formant frequency mapping receives some support, but is not conclusively determined. Finally, a phonemic confusion analysis of the test results is given.

It is concluded that the proposed system shows some promise for superior enhancement performance, but the realization of that promise will require improvements in the modelling of helium speech and in signal acquisition methods.

OBJECTIVE QUALITY MEASURES FOR THE  
DESIGN OF DIGITAL IMAGE TRANSMISSION SYSTEMS

A THESIS

Presented to

The Faculty of the Division of Graduate Studies

by

Bernd Girod

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Electrical Engineering

Georgia Institute of Technology

November, 1980

## SUMMARY

An analytical quality measure is essential for the successful design of digital transmission systems. For images the processes involved in human quality perception are too complicated to be modelled in a mathematically tractable way. In order to obtain useful results we have given up generality and investigated only quality measures for achromatic still images which have been coded by Pulse-Code Modulation (PCM), Pseudo-Random Noise PCM, or Differential PCM (DPCM).

In the first stage of the research a data base of distorted images was generated. These images were the output of a flexible DPCM system which was able to simulate PCM, Pseudo-Random Noise PCM, and DPCM.

The second stage was the subjective rating of the distorted images by a doubly-anchored isometric quality test. The distorted images were presented to human subjects as projected slides. The exposure of the slides had been controlled such that the mapping from the numerical scale in the DPCM-simulation system into the grey scale of the presented pictures was linear.

In the third stage of the research several objective measures were computed for the distorted images and their performance was

compared. The primary methods for the evaluation of an objective measure were its correlation coefficient with the subjective results, the mean square error of objective and subjective measures, and scatter plots of objective versus subjective results.

Evidence has been obtained that the mean squared error is a better quality indicator than it has been commonly assumed. A threshold of the mean squared error below which no quality deterioration can be perceived is important, and estimates for this threshold have been obtained. Frequency-weighted quality measures did not perform well for the class of images under consideration. Another useful measure has the same form as the mutual information between the original and distorted images under the assumption of jointly Gaussian signals.

As a first application, a DPCM system has been optimized with a novel distortion measure. A significant quality improvement has been achieved as compared to the result of a classical design procedure.

## APPENDIX B

### PUBLICATIONS RESULTING FROM GRANT ECS78-17201

#### Publications in Journals

1. T. C. Speake and R. M. Mersereau, "A Note on the Use of Windows for Two-Dimensional FIR Filter Design," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-29, No. 1, Feb. 1981.
2. R. M. Mersereau and T. C. Speake, "A Unified Treatment of Cooley-Tukey Algorithms for the Evaluation of the Multidimensional DFT," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-29, No. 5, Oct. 1981.
3. M. A. Richards, "Application of Deczky's Program for Recursive Filter Design to the Design of Recursive Decimators," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-30, No. 5, Oct. 1982.
4. M. A. Richards, "Helium Speech Enhancement Using the Short-Time Fourier Transform," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-30, No. 6, Dec. 1982.
5. R. M. Mersereau and T. C. Speake, "The Processing of Periodically Sampled Multidimensional Signals," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-31, No. 1, Feb. 1983.

#### Papers in Conference Proceedings

6. R. M. Mersereau, T. H. Joo, and T. C. Speake, "A Comparison of Hexagonally and Rectangularly Sampled Two-Dimensional FIR Filters," 1980 International Conference on Acoustics, Speech, and Signal Processing ( ICASSP-80 ), pp. 729-732.
7. R. M. Mersereau and T. C. Speake, "Generalized Cooley-Tukey Algorithms for Evaluation of Multidimensional Discrete Fourier Transforms," Fourth Int. Conf. on the Analysis and Optimization of Systems, 744-762, Dec. 1980.
8. B. Girod, "Objective Quality Measures for the Design of Digital Image Transmission Systems," ICASSP-81, 1132-1135.
9. P. A. Ramamoorthy, "Two-dimensional Hexagonal Digital Recursive Filters," ICASSP-81, 712-715.

10. M. A. Richards, "A System for Helium Speech Enhancement Using the Short-Time Fourier Transform," ICASSP-81, 1097-1100.
11. T. C. Speake and R. M. Mersereau, "Evaluation of Two-Dimensional Discrete Fourier Transforms via Generalized FFT Algorithms," ICASSP-81, 1006-1009.
12. T. C. Speake and R. M. Mersereau, "An Interpolation Technique for Periodically Sampled Two-Dimensional Signals," ICASSP-81, 1010-1013.
13. R. M. Mersereau and T. C. Speake, "Multidimensional Digital Signal Processing from Arbitrary Periodic Sampling Rasters," Int. Conf. on Digital Signal Processing, Florence, Italy, pp. 93-101, Sept. 1981. Also presented at NSF Joint USA-Italy Workshop on DSP, Portovenere, Italy, Aug. 1981.
14. R. M. Mersereau, E. W. Brown, III, and A. Guessoum, "Evaluation of Multidimensional DFTs on Arbitrary Sampling Lattices," Sixteenth Asilomar Conference on Circuits, Systems, and Computers, Nov. 1982.
15. R. M. Mersereau, E. W. Brown, III, and A. Guessoum, "Row-Column Algorithms for the Evaluation of Multidimensional DFTs on Arbitrary Periodic Sampling Lattices," ICASSP-83, 1264-1267.

APPENDIX C  
SCIENTIFIC COLLABORATORS

Principal Investigators

Dr. Russell M. Mersereau, Associate Professor

Dr. Ronald W. Schafer, Regents' Professor

Co-Investigators

Dr. P. A. Ramamoorthy, Visiting Assistant Professor (6/80-9/80)

Dr. G. A. Shaw, Research Associate (1/80-3/80)

M. A. Richards, doctoral student (1/79-2/82)

T. C. Speake, doctoral student (1/79-2/82)

A. Guessoum, doctoral student (3/82-present)

B. Girod, M. S. student (9/79-12/80)

S. J. Lim, M. S. student (6/82-present)

E. Karlsson, M. S. student (9/81-present)

E. W. Brown, III, undergraduate student (3/82-6/82)

T. H. Joo, undergraduate student (3/79-6/79)

## APPENDIX D

### INVENTIONS AND PATENTS ARISING FROM ECS78-17201

The primary results from this research are the following:

1. An algorithm for enhancing helium speech.
2. Three fast Fourier transform algorithms for evaluating multidimensional DFTs on arbitrary sampling lattices. These correspond to generalizations of the one-dimensional Cooley-Tukey, Winograd, and prime factor FFT algorithms.
3. A procedure for decimating and interpolating multidimensional sequences sampled on arbitrary sampling lattices. In particular the lattice-to-lattice interpolation problem was solved.
4. A series of algorithms was developed for designing and implementing digital filters for hexagonally sampled data.

All of these results are of a theoretical nature and no patent applications have been filed, nor are any such filings anticipated. No hardware of any form was built under this grant.

APPENDIX E

TECHNICAL DESCRIPTION OF PROJECT AND RESULTS

The technical summary of this research is not included with this submission of Form 98A. It will be submitted separately on or about May 15, 1983.

## APPENDIX F

### REPRINTS OF PUBLICATIONS

Included in this appendix are reprints of the publications produced under the complete or partial support of ECS78-17201. It is our belief that copies of all of these for which we have received reprints to date have already been forwarded to NSF, but they are included here for completeness.

E-21-656

**FINAL PROJECT REPORT**

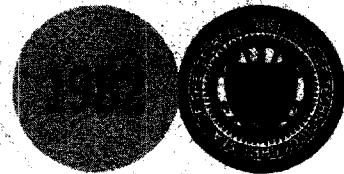
**REPRESENTATION AND PROCESSING OF  
SIGNALS IN TWO-DIMENSIONAL FORM**

**By  
R. M. Mersereau  
R. W. Schafer**

**Prepared for  
NATIONAL SCIENCE FOUNDATION**

**Under  
Grant No. ENG-7817201**

**GEORGIA INSTITUTE OF TECHNOLOGY  
A UNIT OF THE UNIVERSITY SYSTEM OF GEORGIA  
SCHOOL OF ELECTRICAL ENGINEERING  
ATLANTA, GEORGIA 30332**



PLEASE READ INSTRUCTIONS ON REVERSE BEFORE COMPLETING

PART I-PROJECT IDENTIFICATION INFORMATION

1. Institution and Address Georgia Tech Research Institute Georgia Institute of Technology Atlanta, Georgia 30332	2. NSF Program Elec.&Optical Communication	3. NSF Award Number ECS78-17201
	4. Award Period From 1/79 To 9/82	5. Cumulative Award Amount \$122,251
6. Project Title Representation and Processing of Signals in Two-Dimensional Form		

PART II-SUMMARY OF COMPLETED PROJECT (FOR PUBLIC USE)

This research was concerned with finding and exploiting efficient and novel two-dimensional representations of both one-dimensional signals, such as speech and two-dimensional signals, such as photographic images. As such, the research was divided into two parts. A 2-D time-frequency representation for speech was extended and used to enhance the nearly unintelligible speech which results when a submerged diver is breathing a helium rich atmosphere under pressure. Using acoustic models for speech production, a non-linear procedure was developed to alter the frequency spectrum of the distorted speech and a two-dimensional noise stripping procedure was developed to improve the signal-to-noise ratio of the restored speech. These algorithms both improved the quality of the distorted speech, but the noise-stripping procedure did not improve its intelligibility. The second part of this research, which was concerned with efficient representations of two-dimensional signals, considered signals which were sampled on arbitrary periodic sampling lattices. These representations are known to be efficient, but it was not known how signal processing algorithms could be performed using data stored in this format. The current research showed essentially that any signal processing operations that could be performed on one-dimensional signals could be performed on any of these multidimensional representations. These resulting algorithms generally required less computation than when traditional sampling strategies were used. These results are directly applicable not only to problems in multidimensional signal processing, but also to the design of phased-array antennas and beamformers.

PART III-TECHNICAL INFORMATION (FOR PROGRAM MANAGEMENT USES)

1. ITEM (Check appropriate blocks)	NONE	ATTACHED	PREVIOUSLY FURNISHED	TO BE FURNISHED SEPARATELY TO PROGRAM	
				Check (✓)	Approx. Date
a. Abstracts of Theses		X			
b. Publication Citations		X			
c. Data on Scientific Collaborators		X			
d. Information on Inventions		X			
e. Technical Description of Project and Results				X	5/15/83
f. Other (specify) copies of publications		X			
2. Principal Investigator/Project Director Name (Typed) Russell M. Mersereau Ronald W. Schafer	3. Principal Investigator/Project Director Signature			4. Date 4/20/83	

## APPENDIX A

### ABSTRACTS OF THESES COMPLETED

Two doctoral students were supported by this research grant, Mark A. Richards and Theresa C. Speake. Dr. Richards' thesis was completed in February 1982 and was titled "Helium Speech Enhancement Using the Short-Time Fourier Transform." A summary of that thesis appears in this Appendix. Ms. Speake completed all of the work for her thesis but left Georgia Tech before actually writing up the dissertation. Her thesis was to have been titled "Generalized Sampling and Digital Processing of Multi-Dimensional Bandlimited Signals." The primary results of that research are described in many of the publications which are associated with this grant.

In addition to these two doctoral students, one Masters student was partially supported under this grant. The abstract of Mr. Bernd Girod's M.S. thesis entitled "Objective Quality Measures for the Design of Digital Image Transmission Systems" is also included in this appendix.

HELIUM SPEECH ENHANCEMENT USING THE  
SHORT-TIME FOURIER TRANSFORM

A THESIS

Presented to

The Faculty of the Division of Graduate Studies

By

Mark Andrew Richards

In Partial Fulfillment  
of the Requirements for the Degree  
Doctor of Philosophy  
in the School of Electrical Engineering

Georgia Institute of Technology

February, 1982

## SUMMARY

Breathing air at high pressure leads to severe physiological hazards. In particular, the nitrogen present in air can cause nitrogen narcosis and the "bends." Persons who must operate in hyperbaric environments, such as deep sea divers, avoid these problems by breathing an artificial atmosphere, usually composed of helium and oxygen ("heliox"). However, the acoustic properties of hyperbaric heliox differ drastically from those of air, with the result that speech uttered in a heliox atmosphere is virtually unintelligible. The purpose of this thesis is to develop, simulate, and evaluate a new system for enhancing helium speech so as to improve its intelligibility.

The acoustic tube theory of speech is reviewed to indicate how important speech features such as formant frequencies, bandwidths, and amplitudes depend on atmospheric properties. This analysis leads to predictions for the relationship between these speech features in helium speech and normal speech. Two significant points are raised concerning the modeling of helium speech phenomena. The first is a prediction of an increase in formant bandwidths in helium speech, contradicting some previous claims. The second is a discussion of the origins of reported losses in upper formant amplitudes.

The system to be proposed is based on the use of a short-time Fourier transform (STFT) representation of the speech signal. Accordingly, the basic theory of signal processing via the STFT is

reviewed. The choice of sampling rates for the STFT is discussed extensively. Although the basic equations governing this choice are not new, no thorough discussion of their implications has been given. The terminal analog model of speech is used to derive a model for the STFT of speech. This model, which is the discrete equivalent to a recently developed model for the continuous-time STFT, is more general than the traditional quasistationary model in that it allows some variation of the local pitch and vocal tract configuration. However, the simpler quasistationary model is found adequate for this application.

The acoustic analysis and empirical evidence are applied to the speech STFT model to derive an explicit relationship between the STFTs of helium and normal speech, thereby implying a simple enhancement algorithm. The key step in the algorithm is the estimation of the envelope of the STFT in each frame. A simple new algorithm, the "piecewise-linear method", is devised and is shown to be superior to three other envelope estimation techniques in this application. Also, a "spectral subtraction" noise reduction technique is incorporated.

Major computational issues are also discussed. A simple method for early bandwidth reduction is described.

The system is evaluated using formal intelligibility tests. It is found that, in its best configuration, the proposed system operates on a par with the best currently available commercial device. There are some indications that the proposed system may have better performance, but this is not conclusively established. The noise reduction process is found to be detrimental to the intelligibility of the

enhanced speech. Also, the importance of a nonlinear formant frequency mapping receives some support, but is not conclusively determined. Finally, a phonemic confusion analysis of the test results is given.

It is concluded that the proposed system shows some promise for superior enhancement performance, but the realization of that promise will require improvements in the modelling of helium speech and in signal acquisition methods.

OBJECTIVE QUALITY MEASURES FOR THE  
DESIGN OF DIGITAL IMAGE TRANSMISSION SYSTEMS

A THESIS

Presented to

The Faculty of the Division of Graduate Studies

by

Bernd Girod

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Electrical Engineering

Georgia Institute of Technology

November, 1980

## SUMMARY

An analytical quality measure is essential for the successful design of digital transmission systems. For images the processes involved in human quality perception are too complicated to be modelled in a mathematically tractable way. In order to obtain useful results we have given up generality and investigated only quality measures for achromatic still images which have been coded by Pulse-Code Modulation (PCM), Pseudo-Random Noise PCM, or Differential PCM (DPCM).

In the first stage of the research a data base of distorted images was generated. These images were the output of a flexible DPCM system which was able to simulate PCM, Pseudo-Random Noise PCM, and DPCM.

The second stage was the subjective rating of the distorted images by a doubly-anchored isometric quality test. The distorted images were presented to human subjects as projected slides. The exposure of the slides had been controlled such that the mapping from the numerical scale in the DPCM-simulation system into the grey scale of the presented pictures was linear.

In the third stage of the research several objective measures were computed for the distorted images and their performance was

compared. The primary methods for the evaluation of an objective measure were its correlation coefficient with the subjective results, the mean square error of objective and subjective measures, and scatter plots of objective versus subjective results.

Evidence has been obtained that the mean squared error is a better quality indicator than it has been commonly assumed. A threshold of the mean squared error below which no quality deterioration can be perceived is important, and estimates for this threshold have been obtained. Frequency-weighted quality measures did not perform well for the class of images under consideration. Another useful measure has the same form as the mutual information between the original and distorted images under the assumption of jointly Gaussian signals.

As a first application, a DPCM system has been optimized with a novel distortion measure. A significant quality improvement has been achieved as compared to the result of a classical design procedure.

## APPENDIX B

### PUBLICATIONS RESULTING FROM GRANT ECS78-17201

#### Publications in Journals

1. T. C. Speake and R. M. Mersereau, "A Note on the Use of Windows for Two-Dimensional FIR Filter Design," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-29, No. 1, Feb. 1981.
2. R. M. Mersereau and T. C. Speake, "A Unified Treatment of Cooley-Tukey Algorithms for the Evaluation of the Multidimensional DFT," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-29, No. 5, Oct. 1981.
3. M. A. Richards, "Application of Deczky's Program for Recursive Filter Design to the Design of Recursive Decimators," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-30, No. 5, Oct. 1982.
4. M. A. Richards, "Helium Speech Enhancement Using the Short-Time Fourier Transform," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-30, No. 6, Dec. 1982.
5. R. M. Mersereau and T. C. Speake, "The Processing of Periodically Sampled Multidimensional Signals," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-31, No. 1, Feb. 1983.

#### Papers in Conference Proceedings

6. R. M. Mersereau, T. H. Joo, and T. C. Speake, "A Comparison of Hexagonally and Rectangularly Sampled Two-Dimensional FIR Filters," 1980 International Conference on Acoustics, Speech, and Signal Processing (ICASSP-80), pp. 729-732.
7. R. M. Mersereau and T. C. Speake, "Generalized Cooley-Tukey Algorithms for Evaluation of Multidimensional Discrete Fourier Transforms," Fourth Int. Conf. on the Analysis and Optimization of Systems, 744-762, Dec. 1980.
8. B. Girod, "Objective Quality Measures for the Design of Digital Image Transmission Systems," ICASSP-81, 1132-1135.
9. P. A. Ramamoorthy, "Two-dimensional Hexagonal Digital Recursive Filters," ICASSP-81, 712-715.

10. M. A. Richards, "A System for Helium Speech Enhancement Using the Short-Time Fourier Transform," ICASSP-81, 1097-1100.
11. T. C. Speake and R. M. Mersereau, "Evaluation of Two-Dimensional Discrete Fourier Transforms via Generalized FFT Algorithms," ICASSP-81, 1006-1009.
12. T. C. Speake and R. M. Mersereau, "An Interpolation Technique for Periodically Sampled Two-Dimensional Signals," ICASSP-81, 1010-1013.
13. R. M. Mersereau and T. C. Speake, "Multidimensional Digital Signal Processing from Arbitrary Periodic Sampling Rasters," Int. Conf. on Digital Signal Processing, Florence, Italy, pp. 93-101, Sept. 1981. Also presented at NSF Joint USA-Italy Workshop on DSP, Portovenere, Italy, Aug. 1981.
14. R. M. Mersereau, E. W. Brown, III, and A. Guessoum, "Evaluation of Multidimensional DFTs on Arbitrary Sampling Lattices," Sixteenth Asilomar Conference on Circuits, Systems, and Computers, Nov. 1982.
15. R. M. Mersereau, E. W. Brown, III, and A. Guessoum, "Row-Column Algorithms for the Evaluation of Multidimensional DFTs on Arbitrary Periodic Sampling Lattices," ICASSP-83, 1264-1267.

APPENDIX C

SCIENTIFIC COLLABORATORS

Principal Investigators

Dr. Russell M. Mersereau, Associate Professor

Dr. Ronald W. Schafer, Regents' Professor

Co-Investigators

Dr. P. A. Ramamoorthy, Visiting Assistant Professor (6/80-9/80)

Dr. G. A. Shaw, Research Associate (1/80-3/80)

M. A. Richards, doctoral student (1/79-2/82)

T. C. Speake, doctoral student (1/79-2/82)

A. Guessoum, doctoral student (3/82-present)

B. Girod, M. S. student (9/79-12/80)

S. J. Lim, M. S. student (6/82-present)

E. Karlsson, M. S. student (9/81-present)

E. W. Brown, III, undergraduate student (3/82-6/82)

T. H. Joo, undergraduate student (3/79-6/79)

## APPENDIX D

### INVENTIONS AND PATENTS ARISING FROM ECS78-17201

The primary results from this research are the following:

1. An algorithm for enhancing helium speech.
2. Three fast Fourier transform algorithms for evaluating multidimensional DFTs on arbitrary sampling lattices. These correspond to generalizations of the one-dimensional Cooley-Tukey, Winograd, and prime factor FFT algorithms.
3. A procedure for decimating and interpolating multidimensional sequences sampled on arbitrary sampling lattices. In particular the lattice-to-lattice interpolation problem was solved.
4. A series of algorithms was developed for designing and implementing digital filters for hexagonally sampled data.

All of these results are of a theoretical nature and no patent applications have been filed, nor are any such filings anticipated. No hardware of any form was built under this grant.

APPENDIX E

TECHNICAL DESCRIPTION OF PROJECT AND RESULTS

The technical summary of this research is not included with this submission of Form 98A. It will be submitted separately on or about May 15, 1983.

## APPENDIX F

### REPRINTS OF PUBLICATIONS

Included in this appendix are reprints of the publications produced under the complete or partial support of ECS78-17201. It is our belief that copies of all of these for which we have received reprints to date have already been forwarded to NSF, but they are included here for completeness.

## A Note on the Use of Windows for Two-Dimensional FIR Filter Design

THERESA C. SPEAKE AND RUSSELL M. MERSEREAU

**Abstract**—Using a one-dimensional window as a prototype, a two-dimensional window may be formulated having either a square region of support or a circular one. In this paper we compare the effects of using each of these window formulations for 2-D FIR filter design and present formulas for estimating filter order in terms of design specifications, using a Kaiser window as a prototype.

### I. INTRODUCTION

The window function design procedure for 2-D FIR filters is a straightforward extension of the one-dimensional case. To approximate an ideal frequency response  $I(2\pi f_1, 2\pi f_2)$  by an FIR filter, the ideal impulse response  $i(m, n)$  is multiplied by a finite area window array  $w(m, n)$  to produce the filter impulse response  $h(m, n)$ . That is,

$$h(m, n) = i(m, n) w(m, n). \quad (1)$$

The frequency response of the resulting filter,  $H(2\pi f_1, 2\pi f_2)$ , will be a good approximation to  $I(2\pi f_1, 2\pi f_2)$ , whenever  $W(2\pi f_1, 2\pi f_2)$ , the Fourier transform of  $w(m, n)$ , is a good approximation to a two-dimensional impulse function. Two related formulations of window functions are compared here.

The first is the circularly symmetric window formulation described by Huang [1]. These windows have circular regions of support and are formed by sampling rotated one-dimensional continuous window functions,  $w(x)$ , in the two-dimensional plane. Thus,

$$w_H(m, n) = w(\sqrt{m^2 + n^2}). \quad (2)$$

The second type of two-dimensional window is formed as the outer product of two one-dimensional windows:

$$w_G(m, n) = w(m) w(n). \quad (3)$$

This window has a square region of support and is separable in the two independent variables. In the 1-D case, Kaiser [2] has shown that the  $I_0$ -sinh functions defined by

$$w(n) = \begin{cases} \frac{I_0(\alpha \sqrt{1 - (n/R)^2})}{I_0(\alpha)}, & |n| \leq R \\ 0, & |n| > R \end{cases} \quad (4)$$

are extremely good window functions.  $I_0(x)$  is the modified Bessel function of the first kind of zeroth order and  $\alpha$  is an adjustable parameter that specifies a frequency domain trade-off between main lobe width and sidelobe ripple. Using the Kaiser window as a one-dimensional prototype, 2-D windows may then be formed as

Manuscript received June 19, 1979; revised March 3, 1980. This work was supported in part by the National Science Foundation under Grant ECS-7817201 and the Joint Services Electronics Program under Contract DAAG29-76-G-0226.

The authors are with the School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332.

$$w_H(m, n) = \begin{cases} \frac{I_0(\alpha \sqrt{1 - (m^2 + n^2)/R^2})}{I_0(\alpha)}, & \sqrt{m^2 + n^2} \leq R \\ 0, & \sqrt{m^2 + n^2} > R \end{cases} \quad (5)$$

and

$$w_G(m, n) = \begin{cases} \frac{I_0(\alpha \sqrt{1 - (m/R)^2}) I_0(\alpha \sqrt{1 - (n/R)^2})}{I_0^2(\alpha)}, & |m| \leq R \text{ and } |n| \leq R \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

where  $R$  is the radius, in samples, of the desired filter.

The study described in this paper was motivated by the question of which was the better window formulation for the design of 2-D FIR filters. One result of this study is the recognition of some differences between 1-D and 2-D window designs. In addition, design relations have been developed for estimating the minimum filter order required to achieve certain frequency response specifications.

### II. LOW-PASS FILTER DESIGN

For each of the two window function formulations, several filters were designed to approximate circularly symmetric low-pass frequency responses. The ideal frequency response is defined by

$$I(2\pi f_1, 2\pi f_2) = \begin{cases} 1.0, & \sqrt{f_1^2 + f_2^2} \leq f_c \\ 0.0, & \text{otherwise} \end{cases} \quad (7)$$

and the ideal impulse response is

$$i(m, n) = \frac{f_c J_1(2\pi f_c \sqrt{m^2 + n^2})}{\sqrt{m^2 + n^2}} \quad (8)$$

where  $J_1(x)$  is the first-order Bessel function and  $f_c$  is the normalized cutoff frequency with a maximum value of 0.5. The low-pass filter case was chosen for study for several reasons. For one, the impulse response is expressible in closed form. In addition, the specifications of transition bandwidth and percentage overshoot may be clearly defined and measured. Fig. 1 shows a cross-sectional view of the frequency response of a typical filter and defines the frequency specifications. Finally, results obtained for low-pass filters may frequently be generalized for more arbitrary filters.

Using the ideal impulse response and the window functions, 4032 filters were designed with various cutoff frequencies, filter orders, and window function parameters ( $\alpha$ ). The normalized cutoff frequencies of the filters ranged from 0.1 to 0.4 in increments of 0.05. The filter orders, which were always odd, varied from  $5 \times 5$  to  $51 \times 51$ . The filters were restricted to odd orders to ensure zero phase frequency responses and thereby simplify the measurement of passband and stopband ripples. The window function parameter included values from 0.0 (truncation window) to 5.5 in 0.5

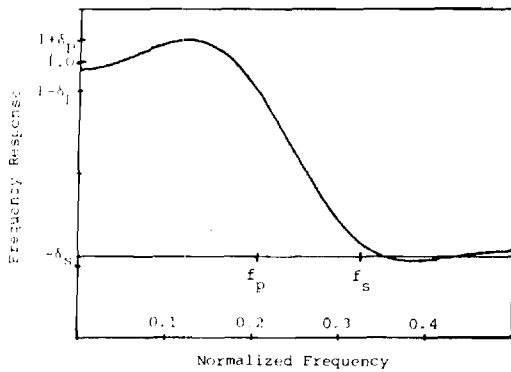


Fig. 1. Low-pass filter specifications for a typical filter.

increments. Half of the filters were designed with the circular window and half with the square window.

The maximum passband ripples, stopband loss, and transition bandwidth in each of the designed filters were measured. To determine the ripple in each filter, a  $256 \times 256$  point 2-D discrete Fourier transform (DFT) was calculated from the filter impulse response and searched to find the maximum value FMAX and the minimum value FMIN. The ripple values were then defined as

$$\begin{aligned} \delta_p &= \text{FMAX} - 1 \\ \delta_s &= -\text{FMIN}. \end{aligned} \quad (9)$$

These values were used in measuring the transition bandwidth, defined to be the difference between the largest stopband frequency  $f_s$  and the smallest passband frequency  $f_p$ . This definition allows for possible deviations from circular symmetry in the filter frequency response. The frequency  $f_s$  was defined as the smallest value of  $\sqrt{f_1^2 + f_2^2}$  such that  $H(2\pi f_1, 2\pi f_2) < \delta_s$  for all  $(f_1, f_2)$  such that  $\sqrt{f_1^2 + f_2^2} > f_c$ . The frequency  $f_p$  was similarly defined as the largest value of  $\sqrt{f_1^2 + f_2^2}$  such that  $H(2\pi f_1, 2\pi f_2) > 1 - \delta_p$  for all  $(f_1, f_2)$  such that  $\sqrt{f_1^2 + f_2^2} < f_c$ .

### III. FILTER ORDER ESTIMATE

Kaiser [3] described a procedure for designing 1-D FIR filters using  $I_0$ -sinh window functions and presented a relation for estimating the required filter order in terms of desired frequency response specifications. The normalized transition bandwidth and amount of passband/stopband ripple in the filter frequency response served as the parameters of the design relation. These same parameters were used to develop similar relations for each of the window functions in the 2-D case.

An unexpected outcome of the measurement process for the 2-D filters was a distinct difference in the amount of passband and stopband ripple for a single filter. For filters designed with the square windows, the passband ripple averaged approximately 25 percent greater than the stopband ripple (not three times greater as given in [4]). For filters designed using circular windows, the passband ripple was about 50 percent greater than the stopband ripple. This is in contrast to the 1-D case in which the passband and stopband ripple are approximately equal. Because of this variation, the filter attenuation was defined to be dependent on the geometric mean  $\sqrt{\delta_p \delta_s}$  of  $\delta_p$  and  $\delta_s$ .

Upon completing the measurement process, it was observed that the filter order was a function of the transition bandwidth and attenuation of the filter, but not of the cutoff frequency. Thus, a design relation having the same form as the 1-D case could be used to estimate the minimum filter order required.

$$N = A(ATT/TB) + B(1/TB). \quad (10)$$

In (10),  $N \times N$  is the number of terms in the filter impulse response,  $ATT = -20 \log_{10}(\sqrt{\delta_p \delta_s})$  is the filter attenuation, and  $TB = f_s - f_p$  is the filter transition bandwidth. By minimizing the squared error over the sum of all the filters in the data base,

$$\sum (N_i - A_i(ATT_i/TB_i) - B_i(1/TB_i))^2, \quad (11)$$

and solving for the coefficients  $A_i$  and  $B_i$ , formulas were obtained to predict filter order for each of the window formulations.

Using the circular window function to design the filters, the following relationship was found to estimate the minimum filter order required to achieve a particular transition bandwidth and attenuation,

$$N_H = \frac{ATT - 7.00}{13.68 TB}. \quad (12)$$

When compared to the previously computed set of filters, this relation predicted the filter order, or was in error by two or less, for 95 percent of the filters.

With the square window function, the filter order relation is

$$N_S = \frac{ATT - 7.99}{13.18 TB}. \quad (13)$$

This relation predicted the correct filter order, or was in error by two or less, for 85 percent of the filters in the data base. (Equation (13) is a corrected version of [4, eq. (14)].) Notice that both of these relations, while having the same form as Kaiser's relation for 1-D filters, also have very similar coefficients. This similarity to the 1-D case was not known in advance.

A design relation for estimating the window parameter ( $\alpha$ ) was also determined by fitting the data. The following expressions work reasonably well. For the circular window,

$$\alpha_H = \begin{cases} 0.56 (ATT - 20.2)^{0.4} + 0.083 (ATT - 20.2), & 60 > ATT > 20.2 \\ 0, & ATT < 20.2 \end{cases} \quad (14)$$

and for the square window,

$$\alpha_S = \begin{cases} 0.42 (ATT - 19.3)^{0.4} + 0.089 (ATT - 19.3), & 60 > ATT > 19.3 \\ 0, & ATT < 19.3. \end{cases} \quad (15)$$

Fig. 2 shows a plot of these relations. Since all of the filters included in the data base had attenuations of 60 dB or less, the window parameter relations are restricted to this range of attenuation.

### IV. CONCLUSIONS

Equations (12) and (13) indicate that for attenuation greater than 34 dB, a somewhat smaller filter order, and thus fewer nonzero impulse response coefficients, will be required when the filter design is accomplished with a circular window. With less attenuation, a slight advantage in filter order is obtained with the square window. However, the implementation of a filter designed with a circular window may require as many as 40 percent fewer nonzero impulse response coefficients. This savings results because the circular window has a round region of support, while the comparable outer product window has a square region of support. If this fact can be exploited in

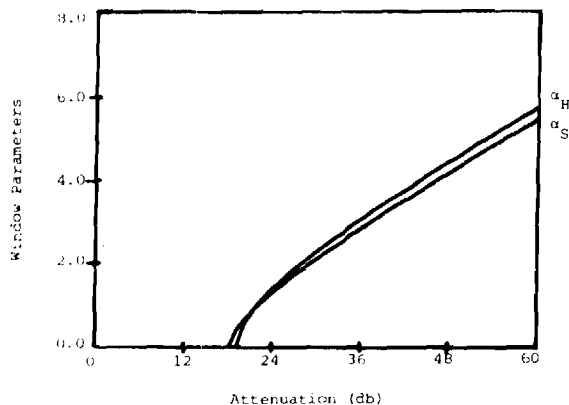


Fig. 2. Window parameters versus filter attenuation.

implementing the filter, it represents a distinct advantage for the rotated windows.

At this point a procedure for designing 2-D FIR filters using windows may be outlined. Depending upon the window function formulation to be used in the design, either (12) or (13) with the desired attenuation and transition bandwidth specifications is used to estimate a necessary window size  $N$ . These equations provide a more accurate estimate of the window size for the circular window. To choose a value for the window parameter, use either (14) or (15), depending on the window type. The window function is then calculated from (5) or (6) with the radius equal to  $(N - 1)/2$  samples. Multiplication of this window function by the ideal impulse response produces the filter impulse response coefficients.

Figs. 3 and 4 show the frequency response plots of two FIR filters designed using window functions. The desired attenuation in both cases was 29 dB and the desired transition bandwidth was equal to 0.175. Fig. 3 shows the frequency response of a filter designed with a circular window having a window parameter  $\alpha = 2.1$  and a radius of 11 samples. Fig. 4 shows the frequency response of a filter designed with a comparable square window. The window size is  $23 \times 23$  samples and the window parameter  $\alpha = 1.9$ . Note that the ripple is circularly distributed throughout the stopband of Fig. 3, while in Fig. 4 it is concentrated about the  $(f_1, f_2)$  axes. This reflects the circular symmetry of the Huang window and the formation of the square window as the outer product of two 1-D windows. In both cases the ripples are dominant near the transition region, which is typical of 1-D filters designed via windows.

In summary, while 1-D filter design using windows is straightforward and well understood, in the 2-D case the procedure is not quite as clear. There are some differences in the performance of the two window formulations considered here. Two-dimensional windows do not seem as well behaved as their

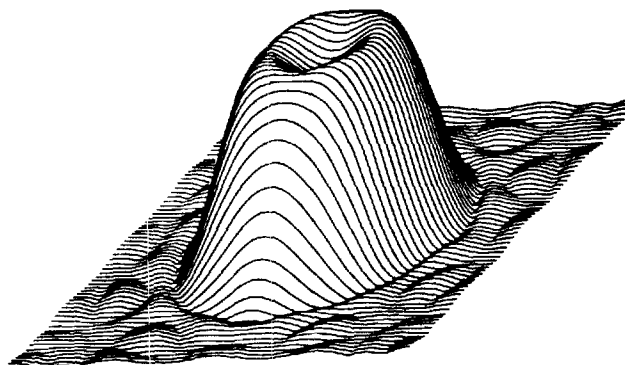


Fig. 3. Frequency response for a filter designed with a circular window.  $ATT = 26.3$  dB.  $TB = 0.15$ .

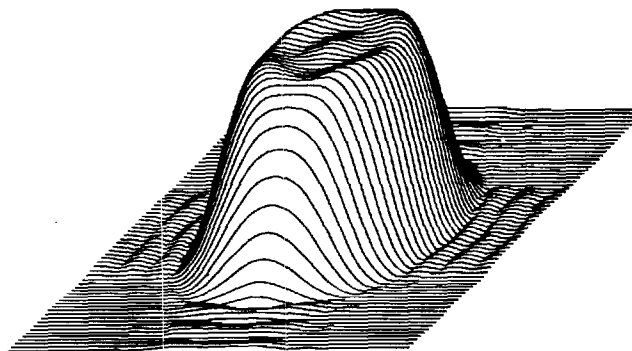


Fig. 4. Frequency response for a filter designed with a square window.  $ATT = 29.6$  dB.  $TB = 0.14$ .

one-dimensional counterparts, and thus, the resulting performance of filters designed by this method is not as predictable with respect to filter orders or the resulting approximation errors. When all factors are taken into account, there seem to be clear advantages to the use of rotated or Huang windows rather than outer product windows. They require fewer nonzero coefficients and seem to be more predictable.

#### REFERENCES

- [1] T. S. Huang, "Two-dimensional windows," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 80-90, Mar. 1972.
- [2] J. F. Kaiser, "Design methods for sampled data filters," in *Proc. 1st Allerton Conf. Circuit Syst. Theory*, Nov. 1963, pp. 221-236.
- [3] —, "Nonrecursive digital filter design using the  $I_0 \sinh$  window function," in *Proc. IEEE Int. Symp. Circuits Syst.*, Apr. 1974, pp. 20-23.
- [4] T. C. Speake and R. M. Mersereau, "A comparison of different window formulations for two-dimensional FIR filter design," in *IEEE Acoust., Speech, Signal Processing Conf. Rec.*, Apr. 1979.

# A Unified Treatment of Cooley-Tukey Algorithms for the Evaluation of the Multidimensional DFT

RUSSELL M. MERSEREAU, SENIOR MEMBER, IEEE, AND THERESA C. SPEAKE

**Abstract**—In this paper the Cooley-Tukey fast Fourier transform (FFT) algorithm is generalized to the multidimensional case in a natural way which allows for the evaluation of discrete Fourier transforms of rectangularly or hexagonally sampled signals or of signals which are sampled on an arbitrary periodic grid in either the spatial or Fourier domain. This general algorithm incorporates both the traditional rectangular row-column and vector-radix algorithms as special cases. This FFT algorithm is shown to result from the factorization of an integer matrix; for each factorization of that matrix, a different algorithm can be developed. This paper presents the general algorithm, discusses its computational efficiency, and relates it to existing multidimensional FFT algorithms.

## I. INTRODUCTION

MANY applications of digital signal processing require the evaluation of discrete Fourier transforms (DFT's) of multidimensional sequences. To cite just a few examples, this calculation is important to the implementation of multidimensional FIR filters, to multidimensional spectral analysis, and to the transform coding of images. These transforms are generally evaluated either by means of a row-column decomposition of the DFT sum, which divides the multidimensional DFT computation into the computation of a number of one-dimensional DFT's [1], or by means of the vector-radix algorithm [2]–[6]. The latter approach, while less well known, offers computational savings which can become quite significant as the dimensionality of the transform is increased.

It is perhaps not surprising that there are other efficient procedures for evaluating multidimensional DFT's. In fact, the row-column decomposition and vector-radix algorithm are simply special cases of a general algorithm which includes many other special cases. Not surprisingly, these alternative FFT algorithms differ in their computational complexity. The algorithm presented in this paper represents a generalization of the Cooley-Tukey [7] algorithm to the multidimensional case. Other special cases of this algorithm can be used to evaluate DFT's of hexagonally sampled sequences [8] or of other periodically sampled sequences [9]. Recent work by Nussbaumer [10] has shown that multidimensional DFT's can be efficiently evaluated using polynomial transforms. This approach, while valid and important, is different from the method presented in this paper and will not be considered in it.

Manuscript received November 4, 1980; revised May 5, 1981. This work was supported in part by the National Science Foundation under Grant ECS-7817201 and by the Joint Services Electronics Program under Contract DAAG29-78-C-0005.

The authors are with the School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332.

It is well known that the efficiency of a 1-D FFT algorithm depends strongly upon the length of the transform,  $N$ . These algorithms become truly efficient only when  $N$  is a highly composite integer. The fundamental theorem of arithmetic states that the prime factorization of any integer is unique except for the ordering of the factors.

The multidimensional counterpart of  $N$  is an integer matrix  $\mathbf{N}$ , called the periodicity matrix. Since the prime factorization of an integer matrix is not unique, we can derive a number of different Cooley-Tukey algorithms whose efficiencies differ. Two specific factorizations lead to the row-column decomposition and the vector-radix algorithm. If sampling strategies other than rectangular sampling are used to sample the signals, different periodicity matrices result, but the resulting DFT summations still have the same structure, and Cooley-Tukey type algorithms can be derived to aid in these calculations also.

This paper is divided into several parts. Section II is primarily concerned with definitions—particularly the definition of the periodicity matrix. The following section presents the general Cooley-Tukey algorithm. Because this derivation is mathematically complicated, in Section IV we present an example, after which we discuss the computational complexity of the algorithm, which is shown to be of the “ $N \log_2 N$ ” form. The relationship between the general FFT and specific algorithms is discussed in Section VI and the paper concludes in Section VII with a discussion of some of the implications of these results.

## II. DEFINITIONS

A one-dimensional sequence  $\tilde{x}(n)$  is periodic if

$$\tilde{x}(n) = \tilde{x}(n + Nr) \quad (1)$$

for all integer values of  $n$  and  $r$  and some positive integer  $N$ , called the period. Such a sequence endlessly repeats itself and any  $N$  consecutive samples define the whole sequence. Similarly, we can say that an  $M$ -dimensional sequence  $\tilde{x}(\mathbf{n})$  is periodic if

$$\tilde{x}(\mathbf{n}) = \tilde{x}(\mathbf{n} + \mathbf{N}\mathbf{r}) \quad (2)$$

for all integer vectors  $\mathbf{n}$  and  $\mathbf{r}$  and some  $M \times M$  integer matrix  $\mathbf{N}$  whose determinant is nonzero. Such a sequence endlessly repeats itself in the  $M$  different directions which are defined by the vectors formed from the columns of  $\mathbf{N}$ . For this reason,  $\mathbf{N}$  is called the *periodicity matrix* of the periodic sequence. While there is no unique “shape” to the set of samples comprising one period of a periodic sequence, a convenient shape

to use for conceptual purposes in the discussions to follow is an  $M$ -dimensional parallelepiped. The number of samples in any period, however, is  $|\det \mathbf{N}|$ , which must be an integer since  $\mathbf{N}$  is an integer matrix. The most commonly encountered periodic sequences are those for which  $\mathbf{N}$  is diagonal. Such sequences are called *rectangularly periodic*.

In (2) and in the remainder of this paper a vector notation has been used to denote multidimensional sequences. Thus, in referring to the  $M$ -dimensional sequence  $\tilde{x}(\mathbf{n})$ ,  $\mathbf{n}$  should be understood to be a column vector whose  $M$  coordinates are integers. A prime (') will be used to indicate the operation of vector or matrix transposition; thus  $\mathbf{n}'$  is a row vector of integers. Lower case boldface letters will be used to denote vectors and upper case boldface letters will be used for matrices.

Any periodic sequence  $\tilde{x}(\mathbf{n})$  with the periodicity matrix  $\mathbf{N}$  can be exactly represented by a set of Fourier series coefficients which will be denoted by  $\tilde{X}(\mathbf{k})$ , where

$$\tilde{x}(\mathbf{n}) = \frac{1}{|\det \mathbf{N}|} \sum_{\mathbf{k} \in J_{\mathbf{N}}} \tilde{X}(\mathbf{k}) \exp [j\mathbf{k}'(2\pi\mathbf{N}^{-1})\mathbf{n}] \quad (3)$$

$$\tilde{X}(\mathbf{k}) = \sum_{\mathbf{n} \in I_{\mathbf{N}}} \tilde{x}(\mathbf{n}) \exp [-j\mathbf{k}'(2\pi\mathbf{N}^{-1})\mathbf{n}]. \quad (4)$$

The sequence of coefficients  $\tilde{X}(\mathbf{k})$  is periodic with periodicity matrix  $\mathbf{N}'$ . The regions  $I_{\mathbf{N}}$  and  $J_{\mathbf{N}}$  denote the set of samples in one period of  $\tilde{x}(\mathbf{n})$  and  $\tilde{X}(\mathbf{k})$ , respectively.

There is a one-to-one correspondence between sequences of finite support confined to  $I_{\mathbf{N}}$  and periodic sequences with periodicity matrix  $\mathbf{N}$  for which  $I_{\mathbf{N}}$  contains the set of samples in one period. Specifically, if  $x(\mathbf{n})$  denotes a sequence with finite support on  $I_{\mathbf{N}}$ , then

$$\tilde{x}(\mathbf{n}) = \sum_{\mathbf{r}} x(\mathbf{n} + \mathbf{N}\mathbf{r}) \quad (5)$$

where  $\mathbf{r}$  varies over all  $M$ -dimensional integer vectors and

$$x(\mathbf{n}) = \begin{cases} \tilde{x}(\mathbf{n}), & \mathbf{n} \in I_{\mathbf{N}} \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

Thus,  $x(\mathbf{n})$  can be completely specified by  $\tilde{x}(\mathbf{n})$ , which in turn can be completely specified by  $\tilde{X}(\mathbf{k})$ , which in turn can be completely specified by  $X(\mathbf{k})$ —the latter being a sequence with finite support on  $J_{\mathbf{N}}$ . The DFT is a concise statement of this relationship:

$$X(\mathbf{k}) = \sum_{\mathbf{n} \in I_{\mathbf{N}}} x(\mathbf{n}) \exp [-j\mathbf{k}'(2\pi\mathbf{N}^{-1})\mathbf{n}], \quad \mathbf{k} \in J_{\mathbf{N}} \quad (7)$$

$$x(\mathbf{n}) = \frac{1}{|\det \mathbf{N}|} \sum_{\mathbf{k} \in J_{\mathbf{N}}} X(\mathbf{k}) \exp [j\mathbf{k}'(2\pi\mathbf{N}^{-1})\mathbf{n}], \quad \mathbf{n} \in I_{\mathbf{N}}. \quad (8)$$

In addition to their interpretation as Fourier series coefficients, the numbers  $X(\mathbf{k})$  can be interpreted in terms of samples of the Fourier transform of the sequence  $x(\mathbf{n})$  or in terms of samples of the Fourier transform of a continuous band-limited signal which has been periodically sampled. This is done explicitly in [9].

In the remainder of this paper we shall consider methods

for evaluating (7). While these algorithms are valid for a general periodicity matrix  $\mathbf{N}$ , some specific periodicity matrices are encountered much more frequently than others. For example, the traditional 2-D DFT, which relates a rectangularly sampled signal to its rectangularly sampled Fourier transform, is characterized by the periodicity matrix

$$\mathbf{N} = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (9)$$

while the hexagonal DFT [8] has the periodicity matrix

$$\mathbf{N} = \begin{bmatrix} N_1 & N_2 \\ N_2 & N_1 \end{bmatrix}. \quad (10)$$

More general (and more exotic) periodicity matrices result when other sampling strategies are used.

### III. THE GENERAL COOLEY-TUKEY ALGORITHM

In this section we shall consider efficient algorithms for the evaluation of (7). FFT algorithms of the Cooley-Tukey type exist whenever the periodicity matrix  $\mathbf{N}$  can be factored into a nontrivial product of integer matrices. This is consistent with the existence condition for a 1-D FFT, which requires that the length of the 1-D DFT be a composite integer. As in the 1-D case, we shall see that the more factors that can be found for  $\mathbf{N}$ , the greater the computational savings.

Any integer matrix  $\mathbf{E}$  for which  $|\det \mathbf{E}| = 1$  is called a *unit matrix*. Clearly,  $\mathbf{E}^{-1}$  is also a unit matrix. Unit matrices are the only integer matrices whose inverses are also integer matrices. If  $\det \mathbf{N}$  is a prime number, we will say that  $\mathbf{N}$  is a *prime matrix*. If  $\mathbf{N}$  is neither a prime nor a unit matrix, we will say that it is *composite*.

Any composite  $2 \times 2$  matrix can be factored into a product of two integer matrices

$$\mathbf{N} = \mathbf{P}\mathbf{Q} \quad (11)$$

where neither  $\mathbf{P}$  nor  $\mathbf{Q}$  is a unit matrix. A constructive proof of this fact is given in the Appendix. It can be hypothesized that composite integer matrices of larger dimensionality can be factored as well. It should be noted that since, in general, matrix products do not commute, the factors in (11) are ordered. It should also be noted that the factorization in (11) is not unique, since

$$\mathbf{N} = [\mathbf{P}\mathbf{E}][\mathbf{E}^{-1}\mathbf{Q}] \quad (12)$$

represents another factorization for any unit matrix  $\mathbf{E}$ .

We shall say that two integer vectors  $\mathbf{m}$  and  $\mathbf{n}$  are *congruent* to one another with respect to the matrix modulus  $\mathbf{N}$  if

$$\mathbf{m} = \mathbf{n} + \mathbf{N}\mathbf{r} \quad (13)$$

for some integer vector  $\mathbf{r}$ . The indexes of the samples of a periodic sequence are thus congruent with respect to the periodicity matrix to the equivalent samples on other periods of the sequence. Specifically, every sample of  $x(\mathbf{n})$  is congruent to a sample from  $I_{\mathbf{N}}$ . We shall use the notation

$$\mathbf{m} = ((\mathbf{n}))_{\mathbf{N}} \quad (14)$$

to denote that vector which is both congruent to  $\mathbf{n}$  and contained in  $I_{\mathbf{N}}$ .

If  $N$  satisfies (11), any vector  $n$  in the region  $I_N$  can be uniquely expressed as

$$n = ((Pq + p))_N \quad (15)$$

where  $p \in I_P$  and  $q \in I_Q$ .  $I_P$  is a set containing  $|\det P|$  vectors and  $I_Q$  is a set containing  $|\det Q|$  vectors. We will have more to say about the exact composition of these two sets shortly. Any pair of vectors—one from  $I_P$  and one from  $I_Q$ —defines a unique vector  $n$  from the set  $I_N$  through (15). (It may be helpful to remember that in the 1-D case,  $q$  is the quotient when  $n$  is divided by  $P$  and  $p$  is the remainder.)

In a similar fashion  $k'$  can be expressed as

$$k' = ((l' + m'Q))_{N'} \quad (16)$$

where  $m \in J_P$  and  $l \in J_Q$ .  $J_P$  and  $J_Q$  are two sets of vectors in the  $k$ -domain whose elements, when combined according to (16), form all of the elements in  $J_N$ .  $J_P$  and  $J_Q$  contain  $|\det P|$  and  $|\det Q|$  members, respectively.

Using (15) and (16), the DFT sum in (7) can be written as

$$X(Q'm + l) = \sum_{p, q} x((Pq + p))_N \cdot \exp[-j(l' + m'Q)(2\pi N^{-1})(Pq + p)]. \quad (17)$$

By expanding the exponential, this sum can be decomposed into two parts.

$$C(p, l) = \sum_{q \in I_Q} x((Pq + p))_N \cdot \exp[-j(l' + m'Q)(2\pi Q^{-1})q] \quad (18a)$$

$$X(Q'm + l) = \sum_{p \in I_P} C(p, l) \exp[-jl'(2\pi N^{-1})p] \cdot \exp[-jm'(2\pi P^{-1})p]. \quad (18b)$$

These relations represent the first level of decomposition of a decimation-in-time Cooley-Tukey FFT algorithm. If  $PQ = QP$ , a similar but different algorithm, corresponding to a decimation-in-frequency FFT, could be analogously derived through the alternative substitutions

$$n = Qp + q \quad (19a)$$

$$k' = l'P + m'. \quad (19b)$$

To understand (18) it is helpful to consider the two equations separately. The sequence  $x((Pq + p))_N$ , when interpreted as a sequence over the vector variable  $q$ , is seen to be periodic with periodicity matrix  $Q$  since

$$((P(q + Qr) + p))_N = ((Pq + p + PQr))_N = ((Pq + p))_N.$$

Thus the summation in (18a) represents a 2-D DFT of the array  $x((Pq + p))_N$  taken with respect to the periodicity matrix  $Q$ . The region of support for this sequence is  $I_Q$ , which must be chosen to be one period of  $x((Pq + p))_N$ , interpreted as a function of  $q$ . A different matrix- $Q$  DFT must be evaluated for each value of the vector  $p$ . This means that  $|\det P|$  such transforms need to be evaluated in all.

The summation in (18b) shows how the outputs of these matrix- $Q$  DFT's should be combined to produce the matrix-

$N$  DFT. The numbers  $C(p, l)$  are first multiplied by the factors  $\exp[-jl'(2\pi N^{-1})p]$  (which are sometimes called twiddle factors), and the products are combined in a series of matrix- $P$  DFT's or *butterflies*. The number of twiddle factor multiplications is  $|\det N|$  and the number of matrix  $P$  butterflies is  $|\det Q|$ . If either  $P$  or  $Q$  is composite, either set of smaller DFT's can be further decomposed.

The vectors

$$n = ((Pq))_N, \quad q \in I_Q \quad (20)$$

form that subset of the region of support which is created by sampling  $I_N$  with the sampling matrix  $P$ . For a fixed value of  $p$ , the samples

$$n = ((Pq + p))_N, \quad q \in I_Q \quad (21)$$

form a coset with respect to this subset. Since each coset is the same size as the subset, there are  $|\det N|/|\det Q| = |\det P|$  cosets in all. The members of any coset are congruent to one another with respect to the modulus  $P$ . The region  $I_P$  should be chosen to consist of one member from each coset— $|\det P|$  elements in all.

The regions  $J_P$  and  $J_Q$  can be chosen similarly. Since the Fourier transform is periodic in  $k$  with periodicity matrix  $N'$  and

$$N' = Q'P'$$

$$k = Q'm + l \quad (22)$$

we see that in the frequency domain  $Q'$  plays a role which is analogous to  $P$  in the spatial domain, and  $P'$  plays a role which is similar to  $Q$ . Thus, we can identify samples of the form

$$k = ((Q'm))_{N'}, \quad m \in J_P \quad (23)$$

as a subset of samples from the frequency domain. The set  $J_P$  should be chosen to consist of a set of  $|\det P|$  vectors which will generate that subset. For fixed values of  $l$ , vectors of the form

$$k = ((Q'm + l))_{N'} \quad (24)$$

then can be sorted into  $|\det Q|$  cosets.  $J_Q$  must be chosen to consist of one member of each coset.

At this point some of these issues can perhaps be clarified through the consideration of a simple example.

#### IV. AN EXAMPLE

As an example to illustrate the general FFT algorithm, consider the evaluation of a  $4 \times 4$  rectangular DFT with the periodicity matrix

$$N = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix} \quad (25)$$

using the factorization

$$P = \begin{bmatrix} 2 & 2 \\ 1 & -1 \end{bmatrix}, \quad Q = \begin{bmatrix} 1 & 2 \\ 1 & -2 \end{bmatrix}. \quad (26)$$

In Fig. 1 we show the region  $I_N$  divided into four cosets by the sampling matrix  $P$ ; a different symbol is used to indicate the members of each coset. Notice that all four cosets have

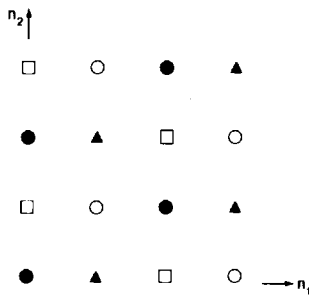


Fig. 1. Spatial domain region of support  $I_N$ , for a  $4 \times 4$  FFT divided into four cosets of samples.

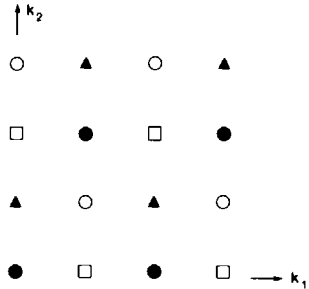


Fig. 2. Frequency domain region of support  $J_N$ , for a  $4 \times 4$  FFT divided into four cosets of samples.

the same geometry when periodically extended. In order to define  $I_Q$ , we need a set of four vectors which will satisfy (20), i.e., we need a set of four vectors which through the use of (20) will span any one of the cosets. One set of vectors which will accomplish this is

$$I_Q = \{(0, 0)', (1, 0)', (2, 0)', (3, 0)'\}. \quad (27)$$

This set is not unique. The set  $I_P$  must be chosen to consist of one member of each coset. Thus one possibility for  $I_P$  is

$$I_P = \{(0, 0)', (1, 0)', (2, 0)', (3, 0)'\}. \quad (28)$$

In Fig. 2 we show the region of support for the DFT,  $J_N$ , divided into four cosets by the frequency domain sampling matrix  $Q'$ . Through an examination of this figure we see that possible choices for  $J_P$  and  $J_Q$  are

$$J_P = \{(0, 0)', (1, 0)', (2, 0)', (3, 0)'\} \quad (29)$$

$$J_Q = \{(0, 0)', (0, 1)', (0, 2)', (0, 3)'\}. \quad (30)$$

Once these four sets have been chosen, a partial flowchart for the algorithm can be drawn, which is done in Fig. 3. Each matrix- $Q$  DFT operates on one of the cosets of the input, shown in Fig. 1, to produce an intermediate array  $C(m_1, m_2)$ . That array is multiplied by the twiddle factors (which for this FFT are all 1) and the results are fed to the matrix- $P$  DFT's. Each of the latter DFT's produces one of the output cosets shown in Fig. 2.

The four outputs of the matrix- $Q$  DFT's can be computed directly from the four inputs. If the inputs are denoted by  $w, x, y,$  and  $z$  and the outputs by  $A, B, C,$  and  $D$ , the direct evaluation of these DFT's corresponds to setting

$$A = w + x + y + z \quad (31a)$$

$$B = w - jx - y + jz \quad (31b)$$

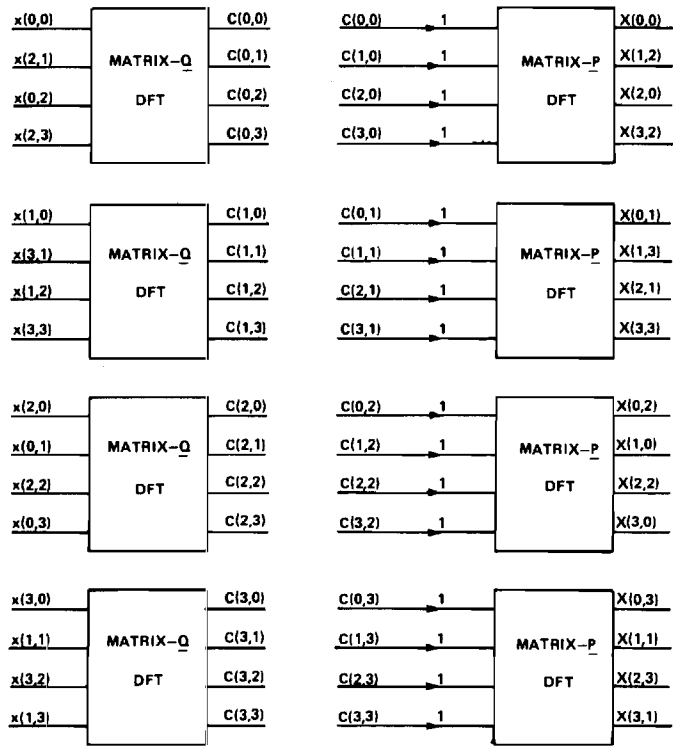


Fig. 3. Flowchart of the  $4 \times 4$  FFT for the example defined by (26).

$$C = w - x + y - z \quad (31c)$$

$$D = w + jx - y - jz. \quad (31d)$$

Alternatively, since  $Q$  is composite, we could use the factorization

$$Q = \begin{bmatrix} 1 & 2 \\ 1 & -2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}. \quad (32)$$

The flowchart of a four point DFT based on this factorization is shown in Fig. 4.

For this particular example the matrix- $P$  DFT's are very similar. If the inputs to these transforms are denoted by  $w, x, y,$  and  $z$  and the outputs are denoted by  $A, B, C,$  and  $D$ , then the direct evaluation results in the same equations as the matrix- $Q$  DFT's which were given in (31). In this example the inputs and outputs may be arranged in such a way that the flowchart in Fig. 4 describes both the matrix- $Q$  DFT's and the matrix- $P$  DFT's.

## V. COMPUTATIONAL COMPLEXITY

If  $C_N$  denotes the computational complexity of a matrix- $N$  FFT algorithm measured in terms of the number of complex multiplications required, then

$$C_N \leq |\det P| C_Q + |\det Q| C_P + |\det N|. \quad (33)$$

The first term represents the number of complex multiplications in  $\det P$  matrix- $Q$  DFT's, the second term represents the contribution from the  $\det Q$  matrix- $P$  DFT's, and the final term represents the contribution from the multiplications by the twiddle factors. This expression is presented as an inequality because in some instances the number of com-

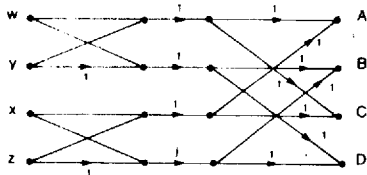


Fig. 4. Flowchart for the evaluation of a matrix- $Q$  DFT using the decomposition in (32), or a matrix- $P$  DFT using the decomposition in (33).

plex multiplications required may be less. This occurs when some of the coefficients in the algorithm reduce to 1,  $-1$ ,  $j$ , or  $-j$ . In the example of the previous section, there were no multiplications due to the twiddle factors, for example. (In fact, for that transform there were no multiplications at all—only complex additions and subtractions.)

This result can be generalized if  $N$  has more than two prime factors. If

$$N = \prod_{i=1}^{\nu} P_i \quad (35)$$

then

$$C_N \leq (\nu - 1) |\det N| + \sum_{i=1}^{\nu} C_{P_i} \prod_{\substack{j=1 \\ j \neq i}}^{\nu} |\det P_j|. \quad (36)$$

Often if  $|\det P_i| = 2, 4, 8$ , or  $16$ , the numbers  $C_{P_i}$  will be zero.

## VI. RELATION OF THE GENERALIZED FFT TO STANDARD MULTIDIMENSIONAL FFT ALGORITHMS

The two common fast Fourier transform (FFT) algorithms for evaluating the DFT's of rectangularly sampled data are the row-column decomposition [1] and the vector-radix algorithm [2]–[5]. Both of these follow from the generalized algorithm presented in this paper for specific factorizations of the periodicity matrix.

The normal (or rectangular) discrete Fourier transform is generally written as

$$X(k_1, k_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x(n_1, n_2) \exp \left[ -j \frac{2\pi}{N_1} n_1 k_1 \right] \cdot \exp \left[ -j \frac{2\pi}{N_2} n_2 k_2 \right] \quad (37)$$

for  $0 \leq k_1 \leq N_1 - 1$ ,  $0 \leq k_2 \leq N_2 - 1$ . In terms of the notation of this paper, this corresponds to

$$N = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (38)$$

$$I_N = \{(n_1, n_2)'; 0 \leq n_1 \leq N_1 - 1, 0 \leq n_2 \leq N_2 - 1\} \quad (39)$$

$$J_N = \{(k_1, k_2)'; 0 \leq k_1 \leq N_1 - 1, 0 \leq k_2 \leq N_2 - 1\}. \quad (40)$$

With the row-column decomposition, this is rewritten as

$$C(n_1, k_2) = \sum_{n_2=0}^{N_2-1} x(n_1, n_2) \exp \left[ -j \frac{2\pi}{N_2} n_2 k_2 \right] \quad (41a)$$

$$X(k_1, k_2) = \sum_{n_1=0}^{N_1-1} C(n_1, k_2) \exp \left[ -j \frac{2\pi}{N_1} n_1 k_1 \right]. \quad (41b)$$

A 1-D DFT is evaluated for each column of the array  $x$  to produce the array  $C$ . Then a 1-D DFT is evaluated for each row of  $C$  to produce  $X$ . If (41) is compared with (18) it is seen that the algorithm is equivalent to the factorization

$$N = \begin{bmatrix} N_1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (42)$$

with

$$I_P = \{(n_1, 0)'; 0 \leq n_1 < N_1\}$$

$$I_Q = \{(0, n_2)'; 0 \leq n_2 < N_2\}$$

$$J_P = \{(k_1, 0)'; 0 \leq k_1 < N_1\}$$

$$J_Q = \{(0, k_2)'; 0 \leq k_2 < N_2\}.$$

The alternative factorization

$$N = \begin{bmatrix} 1 & 0 \\ 0 & N_2 \end{bmatrix} \begin{bmatrix} N_1 & 0 \\ 0 & 1 \end{bmatrix} \quad (43)$$

allows the row transforms to be computed before the column transforms.

When large transforms are to be computed and secondary storage must be used, it is difficult to access data both rowwise and columnwise. For this reason, after the column transforms are computed, a matrix transposition is often performed. Then the original row transforms can be performed as column transforms on the transposed array. The standard algorithm for performing this transposition is through the use of a procedure due to Eklundh [11]. His procedure has  $N \log_2 N$ -type efficiency. This is not surprising if we recognize that Eklundh's procedure is nothing but a degenerate FFT algorithm of the general form presented in this paper, which involves a periodicity matrix which is a unit matrix. The overall row-column algorithm with a transposition stage can be expressed through the factorization

$$N = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & N_1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & N_2 \end{bmatrix}. \quad (44)$$

The two similar matrices involving  $N_1$  and  $N_2$  correspond to column transforms and the other two factors represent the array transpositions.

If  $N_1$  and  $N_2$  in (42) are composite, then additional factorizations can be performed. In the special case that  $N_1$  and  $N_2$  are each powers of two, we can write

$$N = \underbrace{\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \cdots \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}}_{\log_2 N_1 \text{ terms}} \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \cdots \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}}_{\log_2 N_2 \text{ terms}}. \quad (45)$$

This corresponds to the use of a 1-D radix-2 FFT algorithm to evaluate the row and column DFT's.

The extension to higher dimensional row-column algorithms is the obvious one.

If  $N_1$  and  $N_2$  are each divisible by two, it is also possible to perform the factorization

$$\begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} N_1/2 & 0 \\ 0 & N_2/2 \end{bmatrix}. \quad (46)$$

For this factorization one possible choice for the sets  $I_P$ ,  $I_Q$ ,  $J_P$ , and  $J_Q$  is

$$I_P = \{(0, 0)', (0, 1)', (1, 0)', (1, 1)'\}$$

$$I_Q = \{(n_1, n_2)': 0 \leq n_1 < N_1/2, 0 \leq n_2 < N_2/2\}$$

$$J_P = \{(0, 0)', (0, 1)', (1, 0)', (1, 1)'\}$$

$$J_Q = \{(k_1, k_2)': 0 \leq k_1 < N_1/2, 0 \leq k_2 < N_2/2\}.$$

This factorization of  $N$  corresponds to the first stage of decimation for a radix-(2 × 2) vector-radix algorithm. If  $N_1 = N_2 = 2^\nu$ , the complete factorization is

$$N = \underbrace{\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}}_{\nu \text{ terms}}. \quad (47)$$

This result also generalizes in the obvious fashion to higher dimensional transforms.

In [8], Mersereau presented a discrete Fourier transform for hexagonally sampled signals and a fast Fourier transform for evaluating it. That algorithm can be readily seen to correspond to a special case of the algorithm presented here. Specifically, the first stage of the algorithm follows from the factorization

$$N = \begin{bmatrix} 2N & N \\ N & 2N \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} N & N/2 \\ N/2 & N \end{bmatrix} \quad (48)$$

which leads to a vector-radix type algorithm. The regions  $I_P$  and  $I_Q$  for this first stage are

$$I_P = J_P = \{(0, 0)', (0, 1)', (1, 0)', (1, 1)'\}$$

$$I_Q = J_Q = \{(q_1, q_2)': 0 \leq q_1 < 3N/2, 0 \leq q_2 < N/2\}.$$

If  $N$  is a power of two,  $2^\nu$ , we arrive at the more complete decomposition

$$N = \underbrace{\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \cdots \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}}_{\nu \text{ terms}}. \quad (49)$$

The butterflies for all of the stages except the first are alike and contain four inputs and four outputs; those in the first stage contain three inputs and three outputs. A complete flowchart is shown in Fig. 5.

An alternative factorization for the transform is

$$N = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} N & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & N \end{bmatrix}.$$

The latter two factors indicate row and column operations.

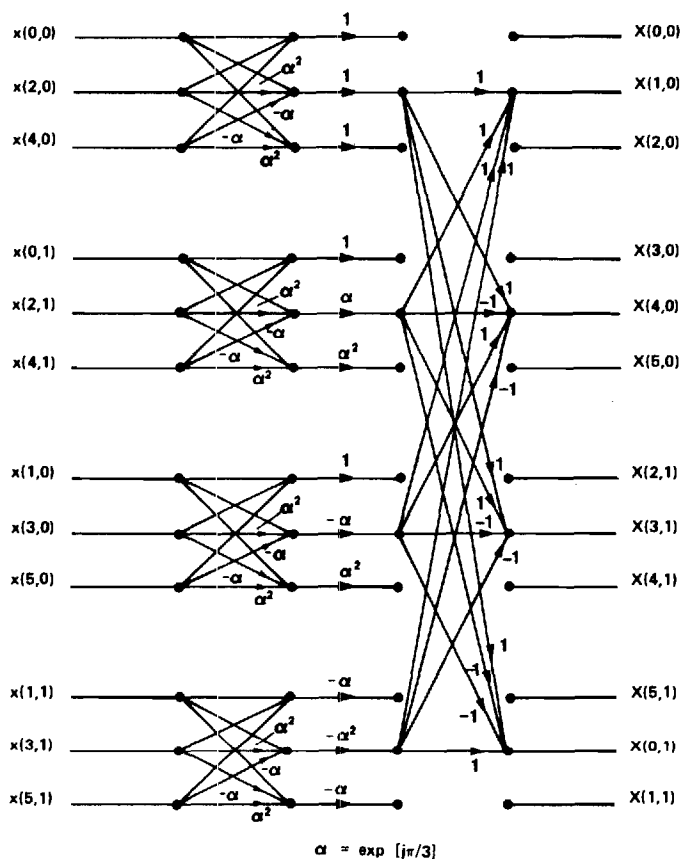


Fig. 5. Flowchart of a vector-radix FFT algorithm for a hexagonal DFT, drawn for the case  $N = 2$ . Only one butterfly in the second stage is shown.

## VIII. DISCUSSION

We have seen in this paper that the specification of a factorization of the periodicity matrix is equivalent to the specification of a multidimensional FFT algorithm. Thus it is possible to postulate many more multidimensional FFT algorithms than are currently known. Furthermore, all of these algorithms are not equally efficient. Since the factorization contains all of the information about an FFT algorithm, we would like to be able to judge the computational efficiency of an algorithm by merely considering the factorization without drawing the flowchart. This problem, however, is not completely solved. For example, it is well known that a (2 × 2) vector radix FFT algorithm requires approximately 25 percent fewer multiplications than the radix-2 row-column FFT algorithm. While the factorization of the periodicity matrix is related to the computational complexity of the resulting algorithm, that relationship is not a simple one.

On the other hand, general statements can be made about the relationship between algorithm complexity and the factorization of the periodicity matrix. It is known that diagonal factors can be implemented simply through row-column type algorithms, or through vector-radix type algorithms. Thus, as a practical matter, it is desirable to first determine the largest diagonal factor (in terms of the magnitude of the determinant) of the periodicity matrix. The remaining non-diagonal factor can be used to design a processing stage that precedes or follows a number of standard rectangular FFT operations.

## APPENDIX

PROOF THAT ANY COMPOSITE  $2 \times 2$  INTEGER MATRIX CAN BE FACTORED

In this Appendix we prove that any  $2 \times 2$  integer matrix that is composite can be factored into a product of nonunit integer matrices. The proof is constructive and consists of two parts. First it is shown that any such matrix can be triangularized; then it is shown that any upper triangular  $2 \times 2$  integer matrix can be factored.

*Theorem:* Any  $2 \times 2$  integer matrix  $M$  which has a nonzero determinant can be written in the form  $M = TE$ , where  $T$  is an upper triangular matrix and  $E$  is a unit matrix.

*Proof:* Assume

$$M = \begin{bmatrix} w & x \\ y & z \end{bmatrix}.$$

If  $y = 0$  choose

$$E = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; \quad T = \begin{bmatrix} w & x \\ y & z \end{bmatrix}.$$

If  $z = 0$  choose

$$E = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}; \quad T = \begin{bmatrix} x & w \\ z & y \end{bmatrix}.$$

If neither  $y$  nor  $z$  is zero, let  $d = \gcd(y, z)$  and set

$$E^{-1} = \begin{bmatrix} z/d & a \\ -y/d & b \end{bmatrix}$$

where  $a$  and  $b$  are chosen to satisfy  $ay + bz = d$ . We are guaranteed that an integer solution exists. For this choice of  $a$  and  $b$ ,  $E^{-1}$  (and hence  $E$ ) is a unit matrix and  $T = ME^{-1}$  is upper triangular.

*Theorem:* Any  $2 \times 2$  upper triangular matrix with a determinant which is a composite integer can be factored.

*Proof:* Let the matrix be denoted  $T$  and written in the form

$$T = \begin{bmatrix} A & B \\ 0 & C \end{bmatrix}.$$

The determinant of this matrix is  $AC$  which, by hypothesis, is factorable. Four cases can be distinguished.

*Case 1:*  $C = \pm 1$ .

In this case  $A$  must be expressible as the product of two factors  $A = DE$  and we have the following factorization.

$$\begin{bmatrix} DE & B \\ 0 & \pm 1 \end{bmatrix} = \begin{bmatrix} D & B - Dy \\ 0 & \pm 1 \end{bmatrix} \begin{bmatrix} E & y \\ 0 & 1 \end{bmatrix}.$$

This factorization will work for any integer choice of  $y$ .

*Case 2:*  $A = \pm 1$ .

In this case we have  $C = DE$  and the following factorization.

$$\begin{bmatrix} \pm 1 & B \\ 0 & DE \end{bmatrix} = \begin{bmatrix} 1 & x \\ 0 & D \end{bmatrix} \begin{bmatrix} \pm 1 & B - xE \\ 0 & E \end{bmatrix}.$$

*Case 3:*  $A$  and  $C$  have a common factor.

Let  $A = A'r$  and  $C = C'r$ . Then

$$\begin{bmatrix} A'r & B \\ 0 & C'r \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & r \end{bmatrix} \begin{bmatrix} A'r & B \\ 0 & C' \end{bmatrix}.$$

*Case 4:*  $A$  and  $C$  have no common factor.

$$\begin{bmatrix} A & B \\ 0 & C \end{bmatrix} = \begin{bmatrix} A & x \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & y \\ 0 & C \end{bmatrix}$$

where  $Ay + Cx = B$ . Since  $A$  and  $C$  are relatively prime, there exists an integer solution to this relation.

This completes the proof.

## ACKNOWLEDGMENT

It is a pleasure to acknowledge the assistance provided by Dr. R. W. Schafer during many discussions leading to the development of this work. We would also like to thank M. A. Richards for his efforts in reading and making corrections to an earlier version of the manuscript.

## REFERENCES

- [1] R. M. Mersereau and D. E. Dudgeon, "Two-dimensional digital filtering," *Proc. IEEE*, vol. 63, pp. 610-623, Apr. 1971.
- [2] D. B. Harris, J. H. McClellan, D. S. K. Chan, and H. W. Schuessler, "Vector radix fast Fourier transform," in *1977 IEEE Int. Conf. Acoust., Speech, Signal Processing Rec.*, 1977, pp. 548-551.
- [3] G. E. Rivard, "Direct fast Fourier transform of bivariate functions," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, pp. 250-252, June 1977.
- [4] E. A. Hoyer and W. R. Berry, "An algorithm for the two-dimensional FFT," *1977 IEEE Int. Conf. Acoust., Speech, Signal Processing Rec.*, 1977, pp. 552-555.
- [5] B. Arambepola, "Fast computation of multi-dimensional discrete Fourier transforms," *Proc. Inst. Elec. Eng. (London)*, vol. 127, pp. 49-52, 1980.
- [6] G. L. Anderson, "A stepwise approach to computing the multi-dimensional fast Fourier transform of large arrays," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 280-284, June 1980.
- [7] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Math. Comput.*, vol. 19, no. 90, pp. 296-301, 1965.
- [8] R. M. Mersereau, "The processing of hexagonally sampled two-dimensional signals," *Proc. IEEE*, vol. 67, pp. 930-949, June 1979.
- [9] T. C. Speake and R. M. Mersereau, "An interpolation technique for periodically sampled two-dimensional signals," in *Proc. ICASSP '81*, 1981, pp. 1010-1013.
- [10] H. J. Nussbaumer and P. Quandalle, "Fast computation of discrete Fourier transforms using polynomial transforms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 169-181, Apr. 1979.
- [11] J. O. Eklundh, "A fast computer method for matrix transposing," *IEEE Trans. Comput.*, vol. C-21, pp. 801-803, July 1972.



Russell M. Mersereau (S'69-M'73-SM'78) was born in Cambridge, MA, on August 29, 1946. He received the S.B., S.M., and Sc.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1969, 1969, and 1973, respectively.

From 1971 to 1973 he was an Instructor in the Department of Electrical Engineering at M.I.T., and from 1973 to 1975 he was with the Research Laboratory of Electronics and Department of Electrical Engineering as a Re-

search Associate. Currently he is an Associate Professor in the School of Electrical Engineering, Georgia Institute of Technology, Atlanta, where he is concerned with the digital signal processing of multidimensional signals.

Dr. Mersereau is a member of Tau Beta Pi, Eta Kappa Nu, and Sigma Xi. He is the corecipient (with D. E. Dudgeon) of the 1976 Browder J. Thompson Memorial Prize from the IEEE and the recipient of the 1977 Research Unit Award from the American Society for Engineering Education Southeastern Section. He received a teaching award from the School of Electrical Engineering in 1978. He was formerly the Associate Editor for Signal Processing of this *TRANSACTIONS* and Technical Chairman of ICASSP '81, and he is currently a member of the ASSP Ad Com and the Technical Committees on Digital Signal Processing and Multidimensional Digital Signal Processing.



Theresa C. Speake was born in Atlanta, GA, on September 18, 1952. She received the B.E.E. degree in electrical engineering from the Georgia Institute of Technology, Atlanta, in 1974, and the M.S. degree in electrical engineering from Stanford University, Stanford, CA, in 1976.

From 1975 to 1976 she was a Senior Engineer at Sylvania Electronics Systems Group, Mountain View, CA. In 1976 she returned to Georgia Tech, where she was awarded a President's Fellowship, to begin working toward the Ph.D. degree in electrical engineering. Since 1977 she has been employed at Georgia Tech as a Research and Teaching Assistant. Her research interests are in the area of multidimensional digital signal processing.

## Application of Deczky's Program for Recursive Filter Design to the Design of Recursive Decimators

MARK A. RICHARDS

*Abstract*—The IEEE Press book *Programs for Digital Signal Processing* includes a program due to Deczky for the least- $p$  design of recursive filters with simultaneous magnitude and group delay constraints. This correspondence outlines a method for generalizing the algorithm to enable it to design linear phase recursive decimators. An example is given.

### I. INTRODUCTION

Martinez and Parks [1] have discussed a class of infinite impulse response (IIR) filters especially well suited for application to sampling rate reduction (also called decimation). These filters have transfer functions in which the denominator polynomial is in powers of  $z^R$ , where  $R$  is the integer decimation factor:

$$H(z) = k \frac{\sum_{l=0}^M n_l z^{-l}}{1 + \sum_{l=1}^Q d_l z^{-Rl}} = \frac{N(z)}{D(z)} \quad (1)$$

Such filters require that only every  $R$ th output sample be computed, eliminating one of the advantages of FIR designs in this application. Martinez and Parks gave an algorithm for the design of filters of the form of (1) (with the additional constraint that all zeros lie on the unit circle) using an equiripple magnitude constraint [1].

This correspondence outlines modifications to the recursive filter design program of Deczky available in the IEEE Press book *Programs for Digital Signal Processing* [2] which enable it to be used for the design of filters of the form of (1). Deczky's program allows constraint of both the magnitude and group delay of the filter, using a weighted least- $p$  error criterion. As  $p$  gets large, the equiripple design is included as a special case. Furthermore, the zeros need not be constrained to the unit circle.

### II. THE METHOD

Deczky's method is described in [3]. The essentials of the technique are first, to express the magnitude and group delay responses of the filter as functions of the radii and angles of the poles and zeros; then to obtain formulas for the partial derivatives of the magnitude and group delay with respect to the radius and angle of a pole or zero; and finally to use these expressions in a Fletcher-Powell algorithm to minimize the approximation error. To generalize the method it is only necessary to recompute the partial derivatives for a denominator in powers of  $z^R$  rather than  $z$ . To this end, substitute  $w = z^R$  into the denominator polynomial in (1):

Manuscript received June 24, 1981; revised November 17, 1981. This work was supported in part by the National Science Foundation under Grant ECS-78172001.

The author was with the School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332. He is now with the Engineering Experiment Station, Georgia Institute of Technology, Atlanta, GA 30332.

$$D(w) = 1 + \sum_{l=1}^Q d_l w^{-l}. \quad (2)$$

Let the roots of  $D(w)$  be denoted in polar coordinates by the set  $\{w_{pj}\} = \{r_{pj}, \theta_{pj}\}$  and those of  $N(z)$  by  $\{z_{oj}\} = \{r_{oj}, \theta_{oj}\}$ . Let  $s = 1/R$ . Then the magnitude function  $A(\omega)$  is given by

$$A(\omega) = K \frac{\prod_{j=1}^{M/2} \{1 - 2r_{oj} \cos(\omega - \theta_{oj}) + r_{oj}^2\}^{1/2} \{1 - 2r_{oj} \cos(\omega + \theta_{oj}) + r_{oj}^2\}^{1/2}}{\prod_{j=1}^{Q/2} \left[ \prod_{i=0}^{R-1} \{1 - 2r_{pj}^s \cos(\omega - s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}\}^{1/2} \{1 - 2r_{pj} \cos(\omega + s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}\}^{1/2} \right]} \quad (3)$$

and the group delay  $\tau(\omega)$  is

$$\begin{aligned} \tau(\omega) = & \sum_{j=1}^{Q/2} \left[ \sum_{i=0}^{R-1} \left\{ \frac{1 - r_{pj}^s \cos(\omega - s(\theta_{pj} + 2\pi i))}{1 - 2r_{pj}^s \cos(\omega - s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}} \right. \right. \\ & \left. \left. + \frac{1 - r_{pj}^s \cos(\omega + s(\theta_{pj} + 2\pi i))}{1 - 2r_{pj}^s \cos(\omega + s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}} \right\} \right] \\ & - \sum_{j=1}^{M/2} \left\{ \frac{1 - r_{oj} \cos(\omega - \theta_{oj})}{1 - 2r_{oj} \cos(\omega - \theta_{oj}) + r_{oj}^2} \right. \\ & \left. + \frac{1 - r_{oj} \cos(\omega + \theta_{oj})}{1 - 2r_{oj} \cos(\omega + \theta_{oj}) + r_{oj}^2} \right\} \quad (4) \end{aligned}$$

where  $\omega$  is in the range  $[-\pi, \pi]$ . Equations (3) and (4) are straightforward generalizations of the expressions in [3], where now each root of  $D(w)$  results in  $R$  roots of  $D(z)$ , and so  $R$  terms in (3) or (4). The minimization is performed using the  $Q$   $w$ -plane poles rather than the  $RQ$   $z$ -plane poles.

The partial derivatives of  $A(\omega)$  and  $\tau(\omega)$  with respect to  $r_{oj}$ ,  $\theta_{oj}$ ,  $r_{pj}$ ,  $\theta_{pj}$ , and  $K$  are required for the minimization. All derivatives with respect to  $r_{oj}$  and  $\theta_{oj}$  are given in [3] and are not repeated here. The new quantities are as follows:

$$\begin{aligned} \frac{\partial A(\omega)}{\partial r_{pj}} = & -sA(\omega) \sum_{i=0}^{R-1} \left\{ \frac{r_{pj}^{2s-1} - r_{pj}^{s-1} \cos(\omega - s(\theta_{pj} + 2\pi i))}{1 - 2r_{pj}^s \cos(\omega - s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}} \right. \\ & \left. + \frac{r_{pj}^{2s-1} - r_{pj}^{s-1} \cos(\omega + s(\theta_{pj} + 2\pi i))}{1 - 2r_{pj}^s \cos(\omega + s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}} \right\} \quad (5) \end{aligned}$$

$$\begin{aligned} \frac{\partial A(\omega)}{\partial \theta_{pj}} = & -sA(\omega) \sum_{i=0}^{R-1} \left\{ \frac{r_{pj}^s \sin(\omega + s(\theta_{pj} + 2\pi i))}{1 - 2r_{pj}^s \cos(\omega + s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}} \right. \\ & \left. - \frac{r_{pj}^s \sin(\omega - s(\theta_{pj} + 2\pi i))}{1 - 2r_{pj}^s \cos(\omega - s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}} \right\} \quad (6) \end{aligned}$$

$$\begin{aligned} \frac{\partial \tau(\omega)}{\partial r_{pj}} = & s \sum_{i=0}^{R-1} \left\{ \frac{(r_{pj}^{s-1} + r_{pj}^{3s-1}) \cos(\omega - s(\theta_{pj} + 2\pi i)) - 2r_{pj}^{2s-1}}{[1 - 2r_{pj}^s \cos(\omega - s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}]^2} \right. \\ & \left. + \frac{(r_{pj}^{s-1} + r_{pj}^{3s-1}) \cos(\omega + s(\theta_{pj} + 2\pi i)) - 2r_{pj}^{2s-1}}{[1 - 2r_{pj}^s \cos(\omega + s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}]^2} \right\} \quad (7) \end{aligned}$$

$$\begin{aligned} \frac{\partial \tau(\omega)}{\partial \theta_{pj}} = & s \sum_{i=0}^{R-1} \left\{ \frac{r_{pj}^s (1 - r_{pj}^{2s}) \sin(\omega - s(\theta_{pj} + 2\pi i))}{[1 - 2r_{pj}^s \cos(\omega - s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}]^2} \right. \\ & \left. - \frac{r_{pj}^s (1 - r_{pj}^{2s}) \sin(\omega + s(\theta_{pj} + 2\pi i))}{[1 - 2r_{pj}^s \cos(\omega + s(\theta_{pj} + 2\pi i)) + r_{pj}^{2s}]^2} \right\} \quad (8) \end{aligned}$$

$$\frac{\partial A(\omega)}{\partial K} = \frac{A(\omega)}{K}. \quad (9)$$

Equation (9), while unchanged, was not given in [3]. Equations (5)–(8) are easily seen to reduce to the forms given in [3] when  $R = 1$ .

### III. THE PROGRAM

Implementing the generalization described above requires substantial modification to Deczky's program as given in [2].

The principally affected subroutines and the nature of the changes are summarized below.

**FUNCT:** Modification of the formulas for the contributions of the poles to the error function and gradient to conform to (2)–(9).

**FMFP:** Modification of the limit on the pole step size so as to maintain stability. It is now required that  $r_{pj}^s < 1$  rather than  $r_{pj} < 1$ .

**FACTOR:** Calculation of the  $z$ -plane pole locations from the  $w$ -plane poles used in the minimization.

**FREDIC:** Modification of the calculation of the contribution of the poles to the various frequency response functions.

### IV. EXAMPLE

Consider the design of a half-band decimator. In this case  $R = 2$  and the ideal frequency response has magnitude 1 and flat group delay for  $|\omega| < \pi/2 = \pi/R$  and zero magnitude with unconstrained group delay elsewhere. The specifications for the modified Deczky method are to have the magnitude approximate 1 for  $|\omega| < 0.48\pi$  and 0 for  $|\omega| > 0.52\pi$ , and be unconstrained in the region  $0.48\pi < |\omega| < 0.52\pi$ . The group delay is to approximate 4 for  $|\omega| < \pi/2$ . Also, the numerator polynomial  $N(z)$  is to be of order  $M = 16$  and the denominator polynomial  $D(w)$  is to be of order  $Q = 6$ . There are then  $M = 16$  zeros and  $RQ = 12$  poles.

Deczky's algorithm requires that the number of zeros on and off of the unit circle and real axis be specified at the outset. Furthermore, the algorithm forces all zeros into linear phase groups whenever the group delay is constrained. In this event, the shape of the group delay curve is determined entirely by the poles, and the zeros serve mainly to control the magnitude response.

In this example, six zeros are on the unit circle (three conjugate pairs), two are on the real axis (one reciprocal pair), and eight (two conjugate reciprocal quartets) are on neither. The unit circle zeros are placed in the stopband to maximize stopband attenuation. Those off of the unit circle control the magnitude response in the passband.

Fig. 1 shows the results obtained using a simultaneous least-square error minimization on both the magnitude and group delay (curves marked  $D1$ ) and on the magnitude only (curves marked  $D2$ ). In the former case the total error used in the minimization is 0.9 times the magnitude error plus 0.1 times the group delay error. For filter  $D1$ , the peak passband and stopband ripples are approximately  $-25$  dB and  $-19$  dB, respectively. The corresponding actual transition band defined by these ripples is  $0.395\pi < |\omega| < 0.565\pi$ . However, the group delay characteristics of this filter are excellent. Fig. 2 shows that the group delay approximation error is never more than 0.083 samples. Filter  $D2$  exhibits peak ripples of  $-39$  dB in the passband and  $-25$  dB in the stopband with a corresponding transition band of  $0.46\pi < |\omega| < 0.52\pi$ . This is a significant improvement in magnitude characteristics, but now the group delay peaks to nearly 17 samples.

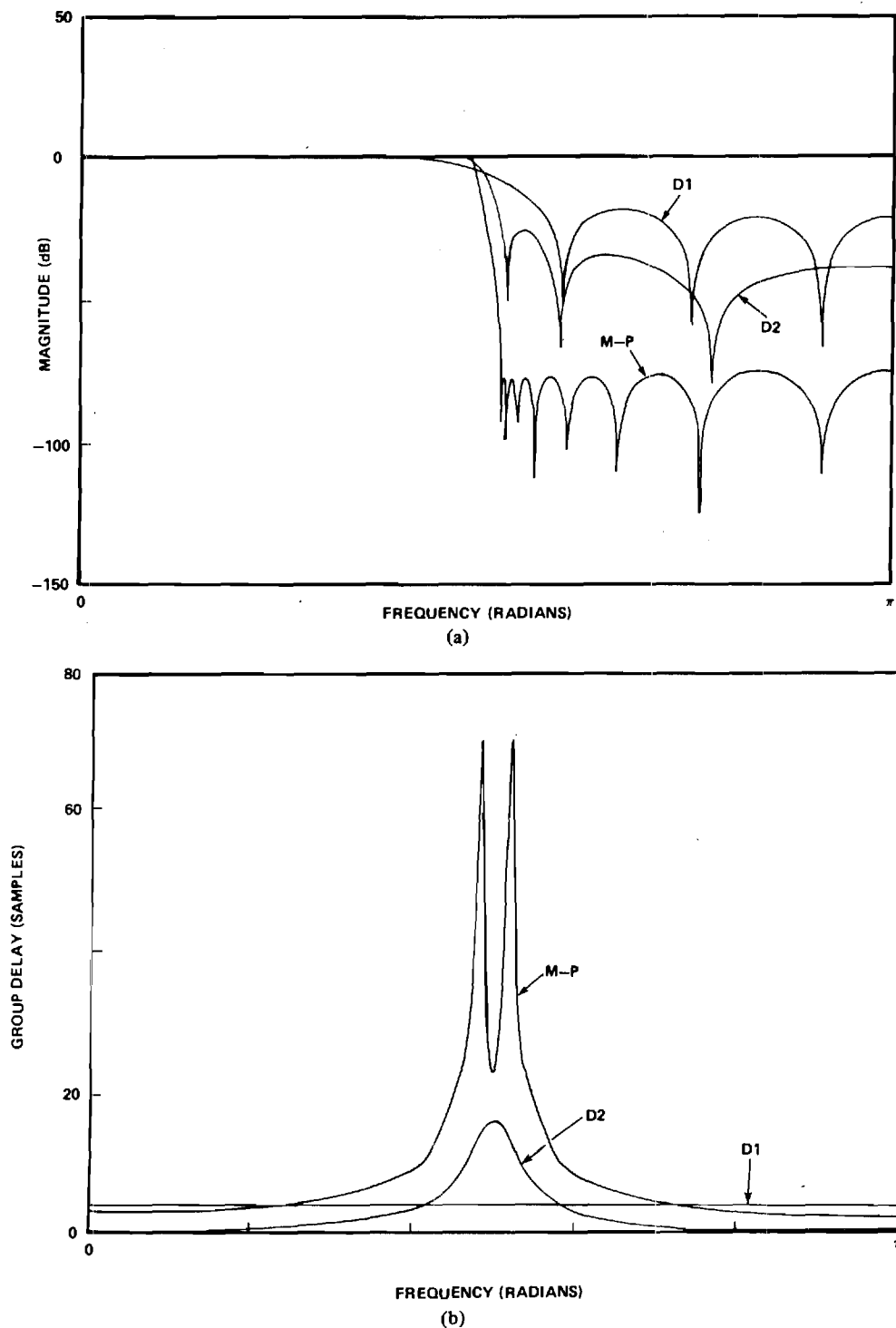


Fig. 1. Comparison of frequency response of half-band decimators designed with the modified Deczky and Martinez-Parks algorithms. "D1" denotes the Deczky filter with simultaneous minimization of both magnitude and group delay error and "D2" the Deczky filter with the same zero complement but minimization of the magnitude error only. "M-P" denotes the Martinez-Parks design. All have 12 poles and 16 zeros. (a) Magnitude response. (b) Group delay.

Filter *D2* uses the same complement of on- and off-unit circle zeros as does filter *D1*. For comparison, Fig. 1 also includes a filter designed using the Martinez-Parks algorithm with the same number of poles and zeros. This algorithm places all of the zeros on the unit circle in the stopband to achieve maximum attenuation. The passband and stopband ripples are approximately -30 dB and -75 dB, and the transition band is exactly  $0.48\pi < |\omega| < 0.52\pi$ . However, the group delay peaks to about 68 samples at the passband edge.

## V. DISCUSSION

The modified Deczky algorithm affords many degrees of freedom in the design of recursive decimators. There is clearly a tradeoff between the quality of the magnitude and group delay approximations. This tradeoff is a function not only of the relative weighting of the magnitude and group delay errors (compare filters *D1* and *D2*), but also of the initial decision of the number and general location of the zeros on and off of the unit circle (e.g., compare the filter *D2* to

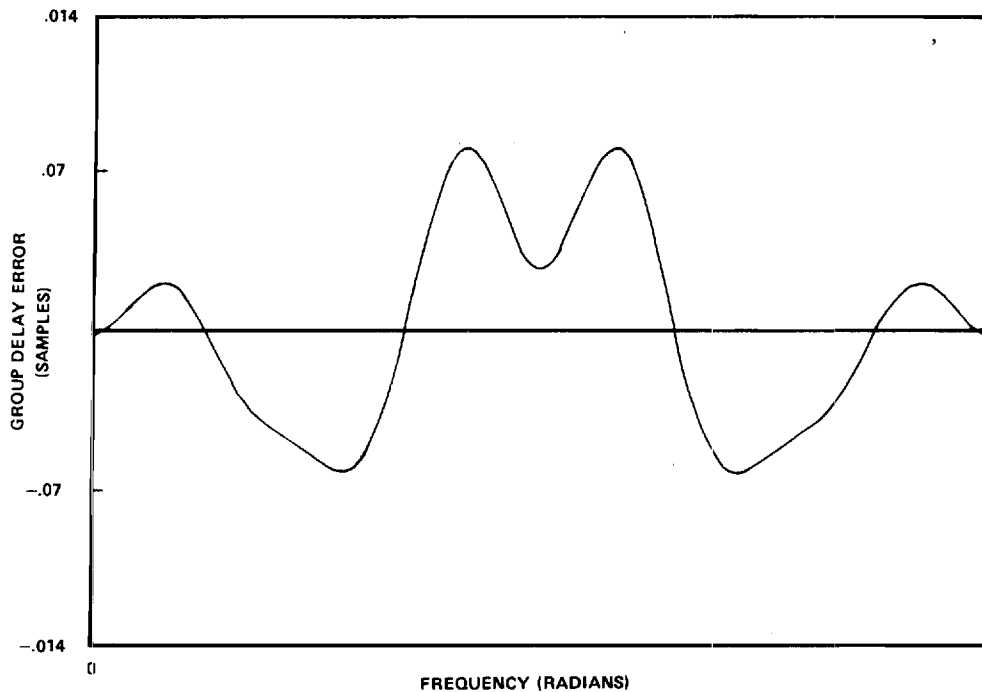


Fig. 2. Group delay approximation error for the modified Deczky filter  $D1$  of Fig. 1.

the Martinez-Parks filter). The only general constraint on the allocation of the zeros is that some should be placed on the unit circle in the stopband to counter the effect of the repeated poles which are due to the decimating structure.

Finally, note that the group delay of the modified Deczky filter  $D1$  is flat for all  $\omega$  although it was constrained only over  $|\omega| < \pi/R$ . This is due to the structure of the denominator polynomial  $D(z)$  and will always be true for a recursive decimator designed by the technique proposed here. This property guarantees the ability to obtain approximately linear phase throughout the transition band and may prove useful in the design of recursive filter banks.

#### REFERENCES

- [1] H. G. Martinez and T. W. Parks, "A class of infinite-duration impulse response digital filters for sampling rate reduction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 154-162, Apr. 1979.
- [2] A. G. Deczky, "Program for minimum- $p$  synthesis of recursive digital filters," in *Programs for Digital Signal Processing*. New York: IEEE Press, 1979, sec. 6.2.
- [3] —, "Synthesis of recursive digital filters using the minimum  $p$ -error criterion," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 257-263, Oct. 1972.

# Helium Speech Enhancement Using the Short-Time Fourier Transform

MARK A. RICHARDS

**Abstract**—Speech produced in a hyperbaric helium-oxygen atmosphere suffers a variety of distortions which render it virtually unintelligible. This paper describes a new system for helium speech enhancement based on a short-time Fourier transform signal representation. The algorithm is robust, allows nonlinear warping of the spectral envelope, and includes provisions for generating the enhanced speech at a reduced sampling rate. Noise reduction by spectral subtraction can be included, and the algorithm is amenable to real-time implementation on an array processor.

A review of helium speech phenomena is included to motivate the design, and a new result for the behavior of formant bandwidths is given. The results of formal intelligibility tests are reviewed. These tests show an improvement in intelligibility from about 40 to 70 percent. The tests also show that the noise reduction scheme is detrimental to intelligibility, but fail to conclusively resolve the importance of a nonlinear formant frequency shift.

## I. INTRODUCTION

THE high pressures involved in deep sea diving prohibit the use of air for breathing, principally because of the effects of nitrogen, which can cause nitrogen narcosis and the "bends." This fact has spurred the development of diving techniques in which the diver breathes an artificial atmosphere, typically a helium-oxygen ("heliox") mixture, at high pressure. Doing so essentially solves the physiological problems, but the "helium speech" produced under these conditions is highly unintelligible, often being described as having a "Donald Duck" quality ("Mickey Mouse" according to [1]). Helium speech is often very noisy because of the various life support machinery and the usually poor acoustic characteristics of the working environment. The correction of helium speech poses an interesting and challenging problem in speech enhancement.

### A. Helium Speech Phenomena

The effects of the high pressure heliox environment are numerous; some are subtle and poorly understood [2]. There are five principal helium speech phenomena which must be considered in the design of a helium speech enhancement system, or "unscrambler." These concern the behavior of the characteristics (center frequency, bandwidth, and amplitude) of the vocal tract resonances, known as formants; the pitch of the speech; and the noise characteristics.

Manuscript received September 8, 1981; revised June 11, 1982 and June 18, 1982. This work was supported in part by the National Science Foundation under Grant ECS-7817201.

The author was with the School of Engineering, Georgia Institute of Technology, Atlanta, GA 30322. He is now with the Engineering Experiment Station, Georgia Institute of Technology, Atlanta, GA 30322.

1) *Formant Frequency Increases*: It is widely accepted that the formant frequencies increase in a nonlinear fashion [1], [3]. If  $F_a$  is the frequency of a given formant in air at sea level, the same formant in a heliox atmosphere will appear approximately at

$$F_h = \sqrt{\alpha^2 F_a^2 + F_o^2} \quad (1)$$

where  $\alpha$  is the ratio of the speed of sound in pressurized heliox to that in air at sea level,  $F_o$  is a complicated function of the properties of the two atmospheres and vocal tract size, and  $F_h$  is the new formant frequency [4], [5].<sup>1</sup> This relation is illustrated in Fig. 1. The linear component of the shift  $\alpha F_a$  is due to the change in the speed of sound, while the nonlinear component is thought to be due largely to increased vibration of the vocal tract walls at low frequencies. For a mixture of 90 percent helium and 10 percent oxygen (appropriate for an operating depth of 350 feet at 10 atmospheres pressure),  $\alpha$  is 2.22 and  $F_o$  is typically about 390 Hz. This large shift of resonant frequencies is the single most important reason for the low intelligibility of helium speech.

2) *Formant Bandwidth Increases*: It has been previously stated [2], [4] that the formant bandwidths do not vary much in helium speech from the values observed in normal speech. The rationale for these statements is not clear. On the contrary, it can be shown that the formant bandwidths will increase significantly in hyperbaric heliox, ranging from a factor of nearly  $\alpha$  on the upper formant bandwidths to more than  $\alpha^2$  on the first formant bandwidth. This result is obtained by analyzing the effects of atmospheric property changes on the vocal tract losses as approximated by Flanagan [6]. Details are given in [5]. Recent preliminary experimental results confirm this phenomenon [7].

An example of the results of the acoustic analysis is given in Fig. 2, which illustrates the approximate contribution of each of the four main vocal loss mechanisms (oral radiation, heat conduction, viscous friction, and glottal loss) to the bandwidths of the first four formants of a particular vocal tract configuration for both a normal atmosphere and a 90-10 percent heliox mixture at 10 atmospheres pressure. The values for the air atmosphere of Fig. 2(a) are taken from the vocal simulations described by Flanagan *et al.* [8]. The values in heliox are derived from those in air using the simplified analysis

<sup>1</sup> Note that this means that the bandwidth of helium speech exceeds that of normal speech by approximately the factor  $\alpha$ , so that correspondingly higher sampling rates are necessary for digital processing.

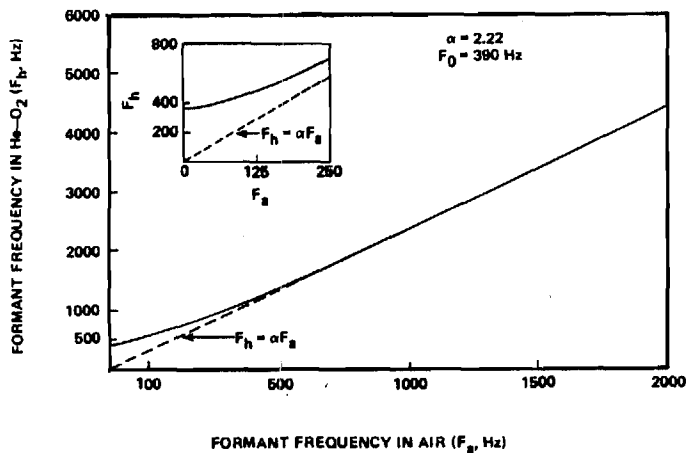


Fig. 1. Translation of formant frequencies in a hyperbaric helium-oxygen atmosphere.

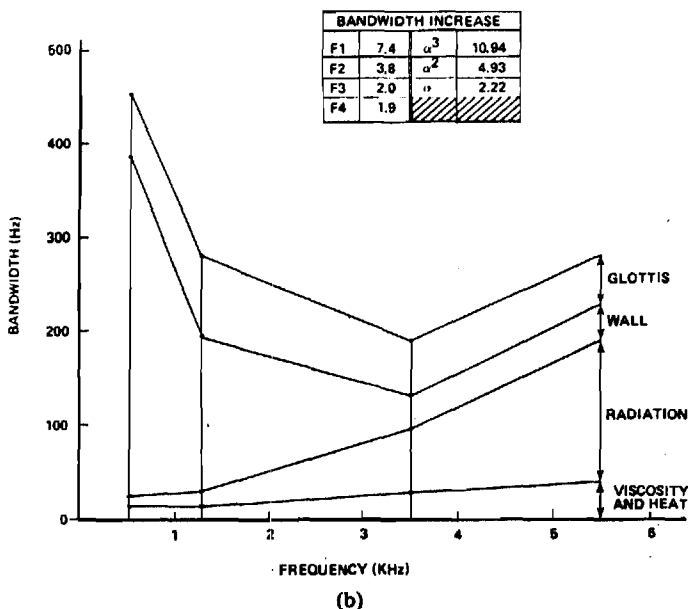
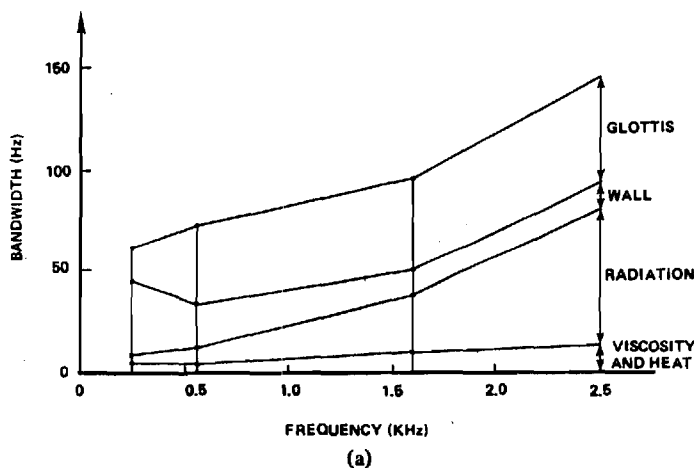


Fig. 2. Relative contributions of various losses to the formant bandwidths. (a) Normal atmosphere, (b) 90-10 percent heliox at 10 ATA. (After Flanagan *et al.* [8].)

of [5]. The inset in Fig. 2(b) gives the ratio of the total formant bandwidths in the two environments.

3) *Attenuation of Higher Formant Amplitudes:* It has been frequently reported that the amplitude of the upper formants

relative to that of the lower formants decreases in helium speech [1], [2], [4], [9]. This effect has not been quantified, and the cause is uncertain. Nakatsui [9] suggested that the glottal pulse spectrum is nearly constant despite the change in atmospheres, so that its high frequency roll-off is responsible for this effect. However, analysis of this suggestion discounts it as a significant factor [5].

Some decrease ( $\approx 4$  dB) is expected in unvoiced sound levels relative to voiced sound levels in the heliox milieu due to the differing physics of the two excitation modes [3]. Since unvoiced sounds tend to have more high frequency content, this effect may have contributed to the observation of lowered high frequency formant amplitudes. Also, it is very likely that at least some of the historically reported loss is in fact due to degraded microphone frequency responses [10]. Thus, although the phenomenon certainly exists, its source and degree are very uncertain.

4) *Pitch Variations:* The pitch of helium speech does not vary much from that of normal speech [4], [11]. Pitch increases of 10-50 percent have been reported in some cases [9]-[12], but even then the increase is not simply predictable from knowledge of the atmosphere. Since small pitch shifts do not degrade speech intelligibility, it will be assumed here that pitch shifts, if any, are insignificant.

5) *High Noise Levels:* Helium speech signals are often contaminated with high levels ( $\approx 0$  dB) of acoustic noise originating from breathing, life support and other machinery noise, and ocean noise. The diver's chamber or helmet may also be significantly reverberant. The noise is generally not white, but rather is lowpass in character. An example noise spectrum is given in Section V.

## B. Previous Enhancement Systems

A wide variety of systems have been proposed for the enhancement of helium speech. These are detailed in [5]; only the general classes of devices are outlined here.

The most common type of system was first described by Stover [13]. These time domain devices segment the helium speech signal; stretch each segment to  $\alpha$  times its original length; and concatenate the modified segments to form the enhanced speech signal. The various implementations differ in whether the segmentation is pitch synchronous or at fixed intervals, and in whether the stretched segments are truncated or allowed to overlap. One variation forms the enhanced speech from the segment autocorrelation [12]. These devices are limited to a linear scaling of the formant frequencies and compress the bandwidths by the factor  $\alpha$ . All can easily be extended to incorporate pitch correction, but only the autocorrelation version incorporates any noise reduction. Most commonly available devices are of this type.

More recent proposals use complex processing of the segments to effect the enhancement. One group of proposals, typified by [14], uses linear predictive analysis to estimate and modify the vocal tract impulse response, while Quick [4] applied homomorphic signal processing techniques. Both of these approaches admit more sophisticated correction of the spectral envelope, and both could be extended to allow pitch modification. Neither incorporated noise reduction.

Other systems which have been proposed include a modified

voice-excited vocoder [15], an analytic signal approach [16], and an analog system due to Copel [17] which splits the signal spectrum into two subbands and heterodynes each by selectable amounts. This latter device has also been produced commercially.

### C. Purpose of the Paper

The purpose of this paper is to describe a new system for helium speech enhancement based on the use of the short-time Fourier transform of the helium speech signal. The system follows the segment-modify-concatenate approach of many previous systems, operating on the Fourier transform of the signal segments to effect the modifications. The system is capable of arbitrary mapping of the spectral envelope, but does not allow pitch modifications. Pitch extraction is not required. With the addition of a speech/silence decision (it is not necessary to distinguish voiced and unvoiced speech), the algorithm can incorporate "spectral subtraction" noise reduction methods in a natural way. It is also simple to halve the system bandwidth prior to output synthesis whenever  $\alpha \geq 2$ , an important computational point.

Formal intelligibility tests are used to evaluate the new system. These tests serve foremost to measure the efficacy of the proposed system, but they also address two other important questions. The first is whether a nonlinear formant frequency shift produces more intelligible enhanced speech than a linear shift. The second is whether the application of the spectral subtraction noise reduction technique increases the intelligibility of the enhanced speech.

## II. THE NARROW-BAND SHORT-TIME FOURIER TRANSFORM OF SPEECH

The short-time Fourier transform (STFT) of a signal  $x(n)$ , evaluated at time  $n$  and radian frequency  $\omega$ , is commonly defined as [18]

$$X(n, \omega) = \sum_{m=-\infty}^{\infty} x(m) h(n-m) e^{-j\omega m}. \quad (2)$$

The function  $h(n)$  is a sliding window which serves to isolate a portion of  $x(n)$  for analysis. When  $h(n)$  is of relatively long duration in time, so that its Fourier transform  $H(\omega)$  is "narrow," the short-time spectrum is referred to as a narrow-band STFT.

It is common to model speech production by a structure like that of Fig. 3. Here a time-varying filter  $T(n, z)$  representing the glottal pulse, vocal tract, and radiation effects is excited by either a quasi-periodic impulse train (for voiced speech) or a white random noise process (for unvoiced speech). Portnoff [19] has studied the STFT of speech modeled by this structure. Simplified versions of his models for the narrowband STFT of speech are given below; the reader is referred to [19] or [5] for derivations.

### A. The Narrow-Band STFT of Voiced Speech

The narrow-band STFT of a voiced speech signal can be well approximated as

$$X(n, \omega) \approx T(n, \omega) U(n, \omega). \quad (3)$$

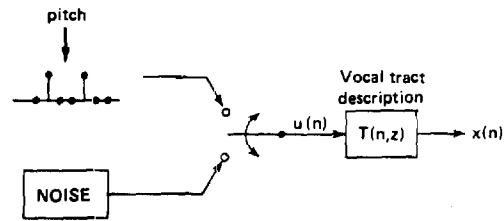


Fig. 3. Model of speech production.

$T(n, \omega)$  is the time-varying frequency response of the vocal tract as shown in Fig. 3.  $U(n, \omega)$  is the STFT of the quasi-periodic impulse train  $u(n)$  which excites the vocal tract filter and is given by

$$U(n, \omega) \approx \begin{cases} \frac{1}{P(n)} H(k\theta(n) - \omega) e^{j[k(\phi(n) + \phi_0) - \omega n]}, & |\omega - k\theta(n)| \leq \omega_h/2 \\ 0, & |\omega - k\theta(n)| > \omega_h/2 \end{cases} \quad (4)$$

for  $k = 0, \dots, P(n) - 1$ .  $P(n)$  is the local pitch period in samples and  $\theta(n) \triangleq 2\pi/P(n)$  is the local fundamental frequency.  $H(\omega)$  is the Fourier transform of  $h(n)$ , and  $\omega_h$  is its approximate two-sided bandwidth. Finally,  $\phi(n)$  is a phase function which is just the running sum of  $\theta(n)$ , and  $\phi_0$  is its initial value

$$\phi(n) = \sum_{m=-\infty}^n \theta(m) \quad (5a)$$

$$\phi(0) = \phi_0. \quad (5b)$$

The relationship of  $T(n, \omega)$ ,  $U(n, \omega)$ , and  $X(n, \omega)$  is depicted in Fig. 4.

Equations (3) and (4), and Fig. 4, illustrate the familiar result that the STFT of voiced speech is a line spectrum. The harmonic structure is due to the near-periodicity of  $u(n)$ . The shape of the lines is determined by the analysis window  $h(n)$ . The line spacing is proportional to the fundamental frequency. The relative amplitudes and phases of the spectral lines are indicative of the values of the vocal tract frequency response  $T(n, \omega)$  at the pitch harmonics.

To derive (3) and (4), it is necessary to assume that the speech signal is quasi-stationary. This implies that the variation with time of both the pitch  $P(n)$  and the vocal tract response  $T(n, \omega)$  are negligible over any interval of the duration of  $h(n)$ . This assumption has proven reasonable most of the time. An expression for the STFT can be developed for the case of small, locally linear variation in time of  $P(n)$ , and arbitrary variation of  $T(n, \omega)$ ; see [5] for details. However, relaxing the quasi-stationarity assumptions quickly complicates the result, with no advantage to this application.

In the context of (3) and (4), the narrow-band requirement is simply that  $\omega_h < \theta(n)$ , so that the spectral lines do not overlap.<sup>2</sup> For a Hamming window of length  $M$ , this is the equiva-

<sup>2</sup>Of course, for commonly used finite windows, the support of  $H(\omega)$  cannot be finite and so there is always some interplay between the harmonics. The two-sided bandwidth  $\omega_h$  is typically taken as the width of the main lobe of  $H(\omega)$ . For a Hamming window,  $\omega_h = 8\pi/M$ , where  $M$  is the window length.

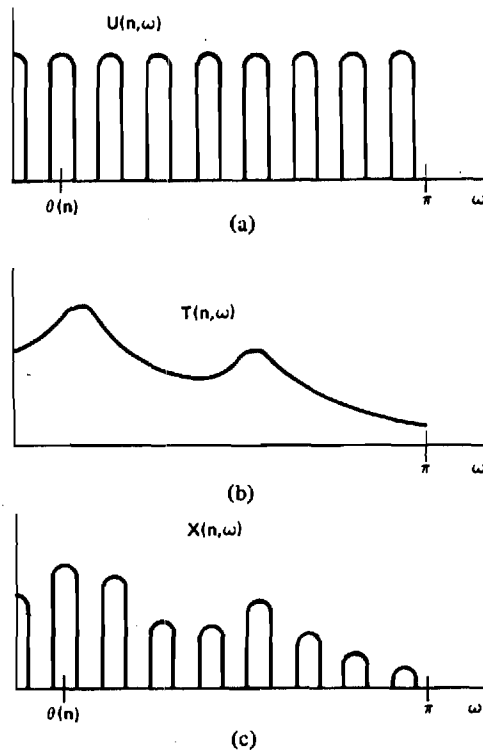


Fig. 4. Idealized narrow-band STFT of voiced speech. (a) STFT of periodic excitation  $u(n)$ , (b) vocal tract frequency response, including glottal and radiation effects, (c) STFT of voiced speech signal  $x(n)$ .

lent to the condition that  $M > 4P(n)$ , where  $M$  is the window length. The importance of the narrow-band requirement is that it makes the value of  $X(n, \omega)$  at a given harmonic proportional to the value of  $T(n, \omega)$  at the same frequency, instead of involving significant contributions from neighboring harmonics as well.

### B. The Narrow-Band STFT of Unvoiced Speech

If the excitation in the speech model of Fig. 3 is taken to be a stationary, zero-mean, white random process with variance  $\sigma_u^2$ , then the STFT will be a zero-mean random process. However, due to the influence of the changing vocal tract filter  $T(n, \omega)$ , it will be neither stationary nor white.

Portnoff [19] has investigated the second-order statistics of  $X(n, \omega)$  in both time and frequency. For the present purpose it is adequate to observe that the expected value of  $|X(n, \omega)|^2$  is

$$E\{|X(n, \omega)|^2\} \approx \frac{\sigma_u^2}{2\pi} |H(\omega)|^2 * |T(n, \omega)|^2 \quad (6)$$

where  $*$  is the convolution operator. That is, the expected value of  $|X(n, \omega)|^2$  is just a smoothed replica of  $|T(n, \omega)|^2$ . Note that, because of the narrow-band assumption, the degree of smoothing is slight. Derivation of (6) requires quasi-stationarity assumptions similar to those described for the voiced case above.

### III. THE RELATION BETWEEN THE STFT OF HELIUM SPEECH AND NORMAL SPEECH

Denote the "envelope" of the magnitude of the STFT by  $A(n, \omega)$ . It is evident from the previous sections that  $A(n, \omega)$

carries the information about the magnitude of the vocal tract frequency response,

$$A(n, \omega) \approx \begin{cases} |T(n, \omega)| & \text{(voiced)} \\ [|T(n, \omega)|^2 * |H(\omega)|^2]^{1/2} & \text{(unvoiced)} \end{cases} \quad (7)$$

and that this envelope must therefore be modified to correct the formant frequencies, bandwidths, and amplitudes. Simultaneously, the underlying harmonic structure must not be disturbed in the case of voiced speech so that the pitch remains constant.

Let the desired normal speech signal be denoted by  $x_a(n)$ . Let its STFT be  $X_a(n, \omega)$ , and the STFT envelope  $A_a(n, \omega)$ . The corresponding quantities for helium speech are  $x_h(n)$ ,  $X_h(n, \omega)$ , and  $A_h(n, \omega)$ . Let the sampling rate used to obtain  $x_h(n)$  be  $T_s$ . The normalized frequency variable  $\omega$  is related to analog domain radian frequencies according to  $\omega = \Omega T_s$ . Define a formant frequency mapping function

$$\xi(\omega) = \sqrt{\alpha^2 \omega^2 + \omega_0^2} \quad (8)$$

where  $\omega_0 = 2\pi F_0 T_s$ , with  $\alpha$  and  $F_0$  as described in Section I. Let  $\xi^{-1}(\omega)$  be the inverse to  $\xi(\omega)$ .

It is shown in [5], and indeed seems reasonable from the foregoing, that the envelope of normal speech can be estimated from that of helium speech as

$$A_a(n, \omega) \approx \begin{cases} C(\omega) A_h(n, \xi(\omega)), & |\omega| \leq \xi^{-1}(\pi) \\ \text{undefined,} & \pi \geq |\omega| \geq \xi^{-1}(\pi). \end{cases} \quad (9)$$

The frequency warping  $\xi(\omega)$  returns the formant frequencies to the correct values. It can also be seen [5] that this mapping reduces the formant bandwidths by very nearly the factor  $\alpha$  at all frequencies. The function  $C(\omega)$  serves to correct the formant amplitudes. The undefined portion of  $A_a(n, \omega)$  occurs because the sampling process discards all information about  $A_h(n, \omega)$  for analog radian frequencies above  $\Omega = \pi/T_s$ .

The function  $C(\omega)$  must be selected empirically due to the lack of detailed knowledge of the various factors affecting the formant amplitudes. This function will generally tend to emphasize high frequencies. The intelligibility of the enhanced speech has been informally found to be insensitive to minor variations in  $C(\omega)$ . Two choices of  $C(\omega)$  which have been used here are shown in Fig. 5. Both give a boost of 6 dB/octave, but differ in the maximum boost and the use of a low frequency cut.

Equation (9) is based on the result that  $|T_a(n, \omega)| \approx |C(\omega) \cdot T_h(n, \xi(\omega))|$  for  $|\omega| < \xi^{-1}(\pi)$  [5]. Thus, it gives precisely the desired result for voiced speech. For unvoiced speech, however, the estimate of (9) does not exhibit the correct smoothing. In particular, from (7) and the assumed form of  $|T_a(n, \omega)|$ , the "actual" value of  $A_a(n, \omega)$  for unvoiced speech, say  $\bar{A}_a(n, \omega)$ , is

$$\bar{A}_a(n, \omega) \approx [|C(\omega) T_h(n, \xi(\omega))|^2 * |H(\omega)|^2]^{1/2} \quad (10)$$

while (9) gives

$$A_a(n, \omega) \approx C(\omega) [|T_h(n, \nu)|^2 * |H(\nu)|^2]^{1/2} \Big|_{\nu=\xi(\omega)}. \quad (11)$$

According to (10), the desired result corresponds to warping and boosting the vocal tract response and then smoothing the

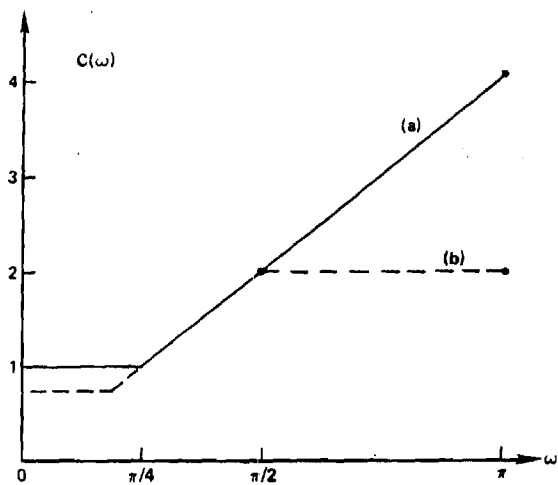


Fig. 5. Two choices for the correction factor  $C(\omega)$ . (a) 12 dB maximum boost with no cut, (b) 6 dB maximum boost with  $-2.5$  dB cut.

spectrum, while (11) smooths the uncorrected spectrum first and then performs the warping and boosting. Block diagrams of the two operations are given in Fig. 6.

In practice, the difference is insignificant. The smoothing of  $A_a(n, \omega)$  is, in effect, less than that of  $\bar{A}_a(n, \omega)$  by roughly the factor  $\alpha$  and is slightly frequency dependent; furthermore,  $C(\omega)$  is excluded from the smoothing in  $A_a(n, \omega)$ . However, since  $H(\omega)$  is narrow band, the degree of smoothing is not great in either case, and the character of the spectra are similar. As for  $C(\omega)$ , it is itself quite smooth to begin with, so that it makes little difference whether it is applied before or after the smoothing. By accepting  $A_a(n, \omega)$  as a reasonable estimate of  $\bar{A}_a(n, \omega)$  for unvoiced speech, (9) can be used to estimate the envelope of the normal speech in both the voiced and unvoiced case, obviating the need to develop a method for distinguishing between them.

Equations (3), (6), and (9) imply the following relation for estimating the STFT of normal speech from that of helium speech

$$X_a(n, \omega) \approx \begin{cases} C(\omega) \frac{A_h(n, \xi(\omega))}{A_h(n, \omega)} X_h(n, \omega), & |\omega| \leq \xi^{-1}(\pi) \\ k(n) \left[ \frac{\pi - |\omega|}{\pi - \xi^{-1}(\pi)} \right], & \pi \geq |\omega| \geq \xi^{-1}(\pi) \end{cases} \quad (12)$$

where the parameter  $k(n)$  is chosen to make  $X_a(n, \omega)$  continuous at  $\omega = \xi^{-1}(\pi)$  in each frame. The potentially undefined region of  $X_a(n, \omega)$  where  $\xi^{-1}(\pi) < |\omega| < \pi$  has been defined by simply tapering the STFT smoothly to zero at  $\omega = \pm\pi$ . Merely setting this region to zero is more likely to result in "musical noise" artifacts [20].

Note that (12) makes no attempt to correct the STFT phase; the phase of the helium speech is simply retained. No information is available to suggest what phase correction is required. Since the ear is known to be relatively insensitive to moderate phase distortion [6], the simple approach of doing nothing has been taken.

Equation (12) is the essence of the enhancement algorithm. It implies that the essential steps are selection of  $C(\omega)$  and of the parameters  $\alpha$  and  $F_0$  of  $\xi(\omega)$ , and then the estimation of the STFT envelope  $A_h(n, \omega)$  in each frame. The details of  $\xi(\omega)$  are determined by the pressure, gas mixture, and speaker [5], and the selection of  $C(\omega)$  has already been discussed. The key operation of envelope estimation is discussed in the next section.

#### IV. ESTIMATING THE SPECTRAL ENVELOPE

Four methods were considered for estimating the STFT envelope  $A_h(n, \omega)$  from  $X_h(n, \omega)$ . The first three are essentially those considered by Makhoul [21], while the fourth is a new algorithm. The discussion implicitly assumes the voiced speech case, but it will be clear that these algorithms will also be suitable for the unvoiced case.

##### A. Smoothing Methods

Two variations on convolutional smoothing of the STFT were tried. In the first,  $|X(n, \omega)|$  was itself smoothed. This approach is similar to the windowed autocorrelation technique of classical spectral analysis and also to ordinary envelope detection schemes. The second version involved smoothing of  $\log [|X(n, \omega)|]$ . This latter case can be viewed as a homomorphic estimate [22], since the logarithm function serves to map the STFT into a space where the envelope and excitation components are additive rather than multiplicative: from (3)

$$\begin{aligned} \log [|X(n, \omega)|] &\approx \log [|T(n, \omega)|] \\ &\quad + \log [|U(n, \omega)|] \\ &= \log [A(n, \omega)] + \log [|U(n, \omega)|]. \end{aligned} \quad (13)$$

The smoothing process is then viewed as an attempt to suppress the excitation component by linear filtering.

In either version, a suitable smoothing filter impulse response was found to be an 11 point Hamming window [when using 512 point discrete Fourier transforms (DFT's)] applied in the frequency domain. This operation could also be implemented as a multiplication in the time or cepstral domain, but for this symmetric low-order filter, direct convolution is more efficient.

##### B. Linear Predictive (LP) Analysis

In this method, the magnitude of the LP estimate of  $X(n, \omega)$  is computed and taken as the envelope estimate. Despite the large bandwidth of helium speech, the number of resonances of the vocal tract are not increased; they are merely spread out according to (1). Consequently, a reasonable predictor order is the same as in the analysis of normal speech, namely 8 to 12. The autocorrelation method of analysis was used because the required autocorrelation lags can be computed from  $|X(n, \omega)|^2$  somewhat more efficiently than a correlation sum can be calculated as required for the covariance method [21].

##### C. The Piecewise-Linear Method

This new algorithm was developed from consideration of the line spectrum structure expected for the STFT of voiced

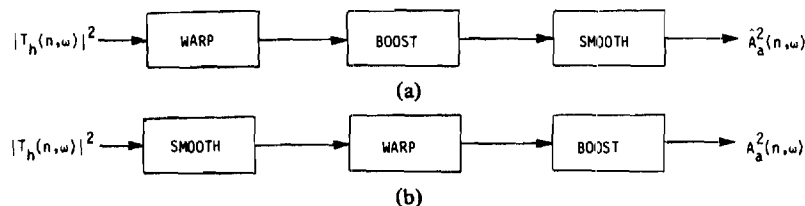


Fig. 6. Estimation of the spectral envelope for unvoiced speech. (a) Ideal method, (b) actual method.

speech as described in (3) and (4). The algorithm uses a peak-picking process to attempt to locate the peak of each STFT spectral line. For the narrow-band STFT, each such peak should provide an accurate measurement of  $A_h(n, \omega)$  at that harmonic frequency. These peaks are then connected by straight line segments to form a piecewise-linear estimate of  $A_h(n, \omega)$ .

The peaks are identified by first creating a list of all local maxima of  $|X_h(n, \omega)|$ . These points form the set of candidates for harmonic line peaks. This list is then edited to eliminate extraneous peaks due to noise and the window sidelobes. The editing begins by assuming that  $|X_h(n, 0)|$  and  $|X_h(n, \pi)|$  are valid points of  $A_h(n, \omega)$ . The editing rule is that the normalized slope connecting a candidate spectral line peak to the previous (in frequency) spectral line peak cannot be less than some minimum negative number  $\epsilon$ .

To quantify the method, let  $P_k \triangleq X(n, \omega_k)$  be the last accepted peak, and  $P_j \triangleq X(n, \omega_j)$  be a candidate for the next line peak, with  $\omega_j > \omega_k$ . Then  $P_j$  is retained as a peak only if

$$\frac{P_j - P_k}{P_k(\omega_j - \omega_k)} > \epsilon. \quad (14)$$

The constraint is intended to keep the spectral envelope from descending too quickly, so as not to follow low-level extraneous features due to noise and the window sidelobes. There is no limit on the rise of the envelope, thereby allowing a quick recovery if a sidelobe or noise peak should be retained. A lower limit for  $\epsilon$  can be computed by calculating the slope of the line connecting the peaks of the main lobe and largest sidelobe of the window transform  $H(\omega)$ , which determines the spectral line shape. For an  $M$  point Hamming window with  $H(0) = 1$ , this is  $-0.035M$ . The parameter  $\epsilon$  would be taken somewhat larger than this number.

Recently Paul [23] has independently described a similar technique for spectral envelope estimation in a vocoder application. His technique differs in that he works with  $\log [|X(n, \omega)|]$  rather than  $|X(n, \omega)|$ , claiming that this gives a "clearer" result, and in using a pitch detector to aid in locating the spectral line peaks. Use of  $\log [|X(n, \omega)|]$  was tested in the helium speech unscrambler, but was found to cause occasional anomalies in the output which were not observed when using  $|X(n, \omega)|$ . No quality improvement was found. Also, introduction of a pitch detector unnecessarily complicates the processor and renders it less robust in this application.

Fig. 7 shows a sample voiced helium speech spectrum,  $\log [|X_h(n, \omega)|]$ , and the envelope estimate  $\log [A_h(n, \omega)]$  obtained from each of the above algorithms. Informal listening tests indicate that all give similar results, with the excep-

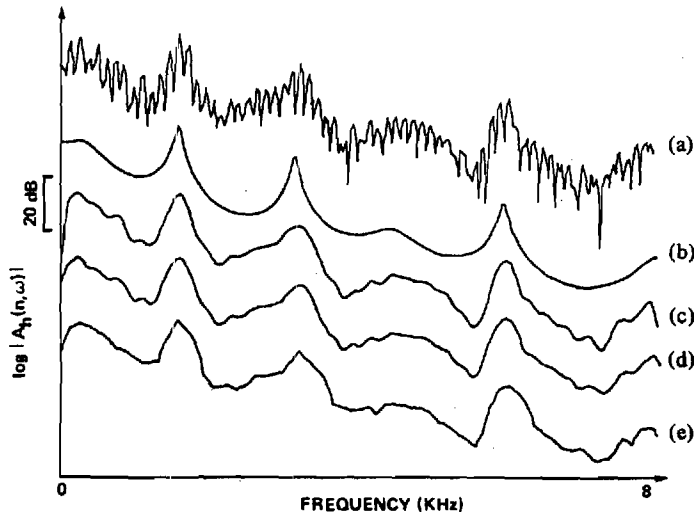


Fig. 7. Estimation of the spectral envelope  $A_h(n, \omega)$ . (a) Original Fourier spectrum; (b) LP estimate, 10th order; (c) smoothed log-magnitude; (d) smoothed magnitude; (e) piecewise-linear estimate.

tion that the linear predictive estimate occasionally gives erroneous high amplitude bursts in the output signal. Table I compares the arithmetic requirements of the four algorithms. Only the operations required to obtain  $A_h(n, \omega)$  from  $|X_h(n, \omega)|^2$  are counted; noise-stripping operations (see Section V) are not included. Clearly, the LP method is by far the most computationally demanding, and the piecewise-linear the least. Therefore, for reasons of simplicity and quality of results, the piecewise-linear envelope estimation technique was adopted.

## V. OTHER ALGORITHM FEATURES

Estimation and correction of the spectral envelope while leaving the fine structure of the STFT intact, as described in the two preceding sections, is the principal operation in the helium speech enhancement algorithm. In this section, three other features of the algorithm are briefly discussed. These are the noise reduction method, the bandwidth reduction method, and the choice of sampling rates in time and frequency.

### A. Noise Reduction by Spectral Subtraction

"Spectral subtraction" noise suppression algorithms are implemented as operations on the STFT of a noisy signal, and so are a natural choice for inclusion in the helium speech unscrambling algorithm. The underlying assumption is that, if  $x_h(n)$  is the sum of a desired signal  $x_d(n)$  and an uncorrelated noise  $x_s(n)$ , then

$$|X_h(n, \omega)|^2 \approx |X_d(n, \omega)|^2 + |X_s(n, \omega)|^2. \quad (15)$$

This comes from viewing the squared-magnitude of the STFT

TABLE I  
COMPUTATIONAL COMPARISON OF ENVELOPE ESTIMATION ALGORITHMS

ALGORITHM	MULTIPLIES	DIVIDES	ADDITIONS	LOG	EXP	( ) <sup>2</sup>	√
LINEAR PREDICTIVE order L (L = 10)	$N(4 \log_2 N + L + 1) + L(2L - 2)$ (24, 244)	$\frac{N}{2} + 1$ (257)	$N(2 \log_2 N + \frac{L}{2} + 1) + L(2L - 2)$ (12, 468)	0	0	$N + L - 1$ (521)	$\frac{N}{2} + 1$ (257)
PIECEWISE LINEAR No. peaks = P (P = 50)	$\frac{N}{2} + P - 1$ (305)	P (50)	$\frac{N}{2} + 3P$ (406)	0	0	0	0
HOMOMORPHIC filter order = 2Q + 1 (Q = 5)	$\frac{N}{2} + 1)(Q + 1)$ (1542)	$\frac{N}{2} + 1$ (257)	$\frac{N}{2} + 1)Q$ (1285)	$\frac{N}{2} + 1$ (257)	$\frac{N}{2} + 1$ (257)	0	0
SMOOTHING filter order = 2Q + 1 (Q = 5)	$\frac{N}{2} + 1)(Q + 1)$ (1542)	0	$\frac{N}{2} + 1)Q$ (1285)	0	0	0	$\frac{N}{2} + 1$ (257)

as a short of "short-time power spectrum." The basic procedure of spectral subtraction is to estimate the noise spectrum  $|X_s(n, \omega)|^2$  and then subtract that estimate from the observed spectrum  $|X_h(n, \omega)|^2$  to develop an estimate of the desired spectrum  $|X_d(n, \omega)|^2$ .

Because (15) is not exact, various heuristic embellishments are used to improve the operation of this process. The version implemented here is adapted from Berouti *et al.* [20]. It estimates  $|X_d(n, \omega)|^2$  according to

$$D(\omega) = |X_h(n, \omega)|^2 - a(n)|X_s(n, \omega)|^2 \quad (16a)$$

$$|X_d(n, \omega)|^2 = \begin{cases} D(\omega), & D(\omega) > b|X_s(n, \omega)|^2 \\ b|X_s(n, \omega)|^2, & D(\omega) \leq b|X_s(n, \omega)|^2 \end{cases} \quad (16b)$$

where  $0 \leq a(n) \leq 5$  and  $b \ll 1$ . The phase of  $X_h(n, \omega)$  is retained to complete the specification of  $X_d(n, \omega)$ . Equation (16a) states that the quantity subtracted from the observed spectrum can range from nothing to a five-times overestimate of the noise spectrum, while (16b) states that the spectral energy is not allowed to go below a "spectral floor" which is a small fraction of the noise spectrum. The spectral floor serves to reduce the "musical noise" effect described in [20], while the parameter  $a(n)$  allows an increase in noise rejection and a reduction in speech distortion over the standard case of  $a(n) \equiv 1$ .

The estimate of the noise spectrum requires a speech/silence decision capability. The STFT is smoothed in frequency and averaged in time during the "silent" intervals to obtain  $|X_s(n, \omega)|^2$ . By using a finite memory averaging process and updating the noise spectrum whenever speech is absent, non-stationary noise characteristics can be tracked.

The subtraction parameter  $a(n)$  is a function of the local signal-to-noise ratio  $SNR(n)$  of the frame being processed. In high SNR frames, the high speech energy masks the noise, so  $a(n)$  is chosen small to minimize distortion. Low SNR frames are more likely to be noise only, so  $a(n)$  is made

large to maximize suppression. In this application,  $a(n)$  was chosen according to

$$\tilde{a}(n) = 0.1 - SNR(n) \quad (17a)$$

$$a(n) = \begin{cases} 0, & \tilde{a}(n) < 0 \\ \tilde{a}(n), & 0 \leq \tilde{a}(n) \leq 5 \\ 5, & \tilde{a}(n) > 5 \end{cases} \quad (17b)$$

where  $SNR(n)$  is in decibels. This function is shown in Fig. 8. The spectral floor parameter  $b$  is usually set to about 0.01, giving 20 dB maximum suppression.

The functional form of (16)-(17) is the same as used in [20], but the particular parameter choices are quite different. This is because the helium speech is much noisier than that considered in [20], and the noise is not at all white. The choices for  $b$  and  $a(n)$  were arrived at primarily through informal experimentation by the author; however, some justification is possible.

The local SNR rarely exceeds 5 dB, and is usually 0 dB and below. Such high noise levels suggest a small value of  $b$  and a large maximum value of  $a(n)$  in order to achieve small residual noise levels. On the other hand, the noise energy is concentrated in the first formant region of frequency, as shown by the smoothed and averaged sample noise spectrum of Fig. 9. This fact demands a rapid reduction of  $a(n)$  to a low minimum value as the SNR becomes positive; otherwise, severe distortion of the speech results.

This algorithm has been informally judged to be moderately effective in reducing noise and improving the quality of the enhanced helium speech. No formal quality tests have been done. However, as will be seen in Section VII, the algorithm appears to be slightly *harmful* to the intelligibility of the enhanced speech. There are two likely reasons for this behavior.

First, the parameter choices described above result in a swing from no subtraction to severe subtraction with a change of only a few dB in local SNR. In essence, the algorithm is unable to separate low-level (typically unvoiced) speech from the

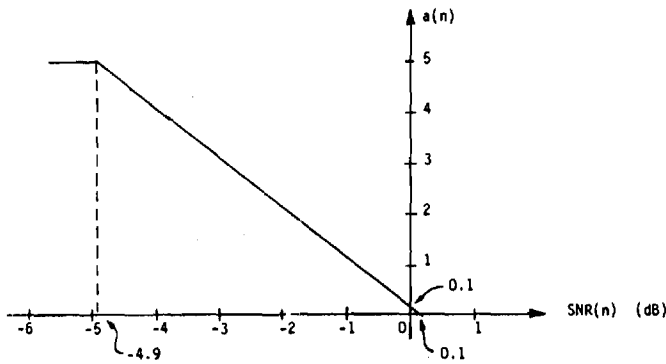


Fig. 8. Variation of spectral subtraction parameter  $a(n)$  with local signal-to-noise ratio SNR( $n$ ).

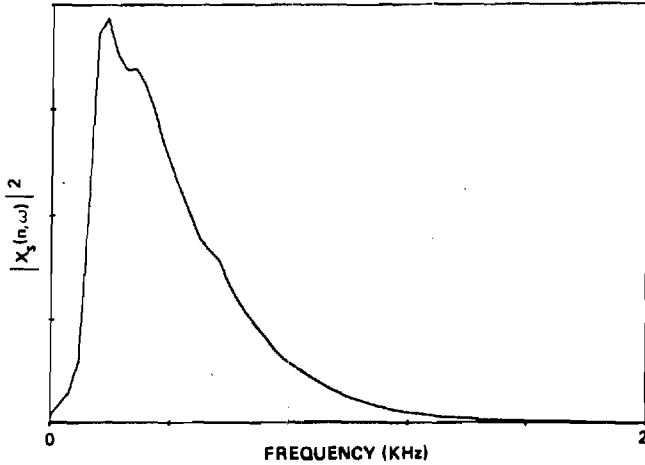


Fig. 9. Sample smoothed and averaged noise spectrum estimate.

noise, and so allows it to be suppressed with the noise. Only the relatively strong voiced passages, in general, pass through the noise reduction process. The slight information about unvoiced sounds present in the noisy speech is lost altogether. Second, the problem of adequately updating the noise spectrum estimate has not been solved. This results in occasional poor estimates of the local SNR. When coupled with the critical behavior of  $a(n)$ , this phenomenon leads to occasional noise bursts (SNR estimate too high), or suppression of significant speech sounds (SNR estimate too low). These effects are shown in Section VI.

### B. Bandwidth Reduction

One of the consequences of the spectral envelope compression of (12) is that the effective bandwidth of the signal represented by  $X_a(n, \omega)$  is less than that of  $X_h(n, \omega)$  by about the factor  $\alpha$ . Whenever  $\alpha > 2$ , the system bandwidth can be halved *prior* to generation of the enhanced speech  $x_a(n)$  at a significant computational savings. The remaining decimation must be done using conventional techniques [24].

To accomplish this decimation, assume that  $X_a(n, \omega)$  would be represented without decimation by the  $N$  point DFT  $X_a(n, k)$ ,  $k = 0, \dots, N-1$ , and that  $\alpha > 2$ . The inverse DFT (IDFT) of  $X_a(n, k)$  (in  $m$ ) is the sequence  $x_a(n, m)$ ,  $m = 0, \dots, N-1$ . Because of the spectral envelope compression involved

in forming  $X_a(n, \omega)$ ,  $X_a(n, \omega)$  is approximately band limited to  $|\omega| \leq \pi/2$ .  $X_a(n, k)$  is therefore approximately zero for  $k = N/4, \dots, 3N/4$ .

Now form an  $N/2$  point DFT representation of  $X_a(n, \omega)$  as follows:

$$\bar{X}_a(n, k) = X_a(n, k) + X_a\left(n, k + \frac{N}{2}\right), \quad k = 0, \dots, \frac{N}{2} - 1. \quad (18)$$

Only one term on the right-hand side of (18) is nonzero for each  $k$  due to the band limiting, so that

$$\bar{X}_a(n, k) = \begin{cases} X_a(n, k), & k = 0, \dots, \frac{N}{4} - 1 \\ 0, & k = \frac{N}{4} \\ X_a\left(n, k + \frac{N}{2}\right), & k = \frac{N}{4} + 1, \dots, \frac{N}{2} - 1. \end{cases} \quad (19)$$

This process is depicted in Fig. 10. Using elementary properties of the DFT, the inverse transform of  $\bar{X}_a(n, k)$  is given by

$$\bar{x}_a(n, m) = x_a(n, 2m), \quad m = 0, \dots, \frac{N}{2} - 1. \quad (20)$$

That is, the enhanced output has been decimated by a factor of 2. If  $x_h(n)$  was obtained at a sampling interval  $T_s$ , then the enhanced speech when using this technique will correspond to a sequence obtained by sampling at an interval of  $2T_s$ . This procedure allows the substitution of an  $N/2$  point IDFT and vector add for an  $N$  point IDFT and vector add, and eliminates computation of  $X_a(n, k)$  for  $k = N/4, \dots, 3N/4$  in the processing of each data frame.

### C. Time and Frequency Sampling Rates

It is not necessary to compute  $X_h(n, \omega)$  for all  $n$  and  $\omega$ ; instead, it is computed only every  $R$  samples in time and at  $N$  frequency values  $\omega_k = k2\pi/N$ ,  $k = 0, \dots, N-1$ . Obviously, it is desirable to choose  $R$  as large and  $N$  as small as possible in order to minimize the overall computational load.

The minimum sampling density is determined by the need to avoid aliasing in the sampled STFT. Generally speaking, the sampling density must be increased when spectral modifications of the type of (12) are to be used over the density required in the absence of modifications. Portnoff [18] has given equations from which the minimum sampling rates can be deduced once the form of the spectral modification is given. An extensive discussion of the implications of these equations and of their application to the helium speech enhancement algorithm is given in [5]. It is shown there that the sampling density can be maintained at the rate used in the absence of spectral modifications in this application.

Doing so clearly allows some aliasing of the STFT. However, the spectral modification performed is relatively slowly vary-

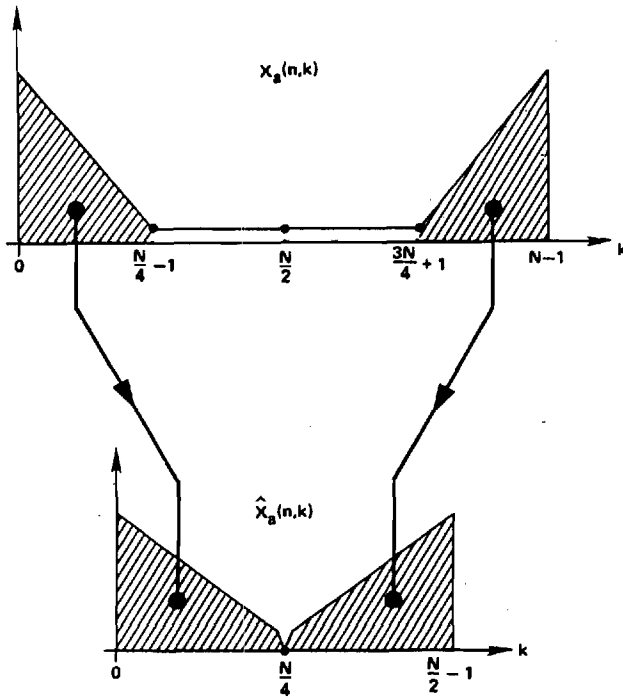


Fig. 10. Formation of the reduced bandwidth DFT representation of  $X_a(n, \omega)$ .

ing in both time and frequency, so that the aliasing is not great. Furthermore, because of the poor quality of helium speech, the aliasing artifacts are negligible compared to the principal distortions which remain even in the enhanced speech. In a higher quality application, it would be essential to increase the sampling density.

The actual choice of rates is now a function only of the window. The interval between frames  $R$  must be chosen as the greatest number so that the succession of overlapped windows adds to a constant

$$\sum_{m=-\infty}^{\infty} h(nR - m) = \text{constant}, \quad \forall m. \quad (21)$$

The number of frequency samples  $N$  is chosen as the least power of 2 (for convenience in using the FFT) such that  $N \geq M$ , where  $M$  is the length of the window  $h(n)$ .

#### D. Array Processor Implementation

Most of the operations required by the enhancement algorithms are vector operations, e.g., windowing, DFT computation, scalar product, vector modulus, and vector sum. These operations are amenable to efficient realization on an array processor. The principal exception is the piecewise-linear envelope estimation, which has been seen to require little computation. Real-time operation of the algorithm therefore seems feasible.

In fact, a real-time implementation of a version of this algorithm has recently been reported [7]. This version uses a 14.706 kHz sampling rate. A 512 point triangular window with 50 percent overlap is used, giving a frame length of 34.8 ms, and an interframe interval of 17.9 ms. The latter value is

the time available for processing a frame. The noise reduction technique was not included in this implementation.

#### VI. AN EXAMPLE

The operation of the STFT based helium speech enhancement algorithm can be illustrated with the aid of the spectrograms of Fig. 11. The helium speech from which these spectrograms were obtained was sampled at a rate of 16 kHz. The analysis window  $h(n)$  was a 512 point (32 ms) Hamming window. The corresponding time sampling interval  $R$  for  $X_h(n, \omega)$  is then 256 samples (16 ms), and the number of frequency samples  $N$  is 512.

A rectangular window would allow a larger value of  $R$ , but "discontinuities" then occur at the frame boundaries which give rise to a clearly perceptible clicking noise. This problem persists in lesser degree when using other windows which are tapered only over part of their length (e.g., trapezoidal or cosine-tapered).

The narrow-band requirement was that the pitch harmonics be resolved so as to allow an accurate envelope estimate. The bandwidth of the Hamming window may be taken as the width of its main lobe, which is  $4/(512T_s)$ , or 125 Hz for the 512 point window. This window completely resolves the spectral lines for pitch frequencies down to 125 Hz, and adequately resolves them to well below 100 Hz.

The spectral warping parameters must be estimated from knowledge of the atmospheric conditions (composition, pressure, etc.) and the ideal gas laws [5] or from any measured data which may be available. In this example, very little concrete information was available. The values selected were  $\alpha = 2$  and  $F_o = 0$  Hz; these values gave a reasonable sounding result.

The bandwidth reduction method was used, so that the output sequence was interpreted as samples of a signal taken at a rate of 8 kHz.

Fig. 11 shows narrow-band spectrograms of the utterance "where's the mercury vapor light?" Darkness of the spectrogram is proportional to the energy at that time and frequency. All three plots are to the same scales. Fig. 11(a) is the original helium speech. The formant tracks are readily evident. The individual pitch harmonics show as the striations in the spectrogram. The noise spectrum of Fig. 9 is visible as the dark band across the bottom of the spectrogram.

Fig. 11(b) and 11(c) show the enhanced speech without and with the noise reduction included, respectively. Compression of the spectral envelope while maintaining the pitch harmonic structure is clearly seen in both results. Fig. 11(c) further shows suppression of the noise in periods of speech inactivity. The two principal flaws in the noise reduction process are also evident in Fig. 11(c). A noise burst can be seen at the very end of the sentence, while unwanted suppression of speech occurs at approximately 0.4 s and 1.1 s [compare to Fig. 11(b)].

#### VII. EVALUATION OF THE ALGORITHM

Formal subjective intelligibility tests were undertaken to evaluate the effectiveness of the enhancement algorithm. Also sought were answers to questions of whether the nonlinear

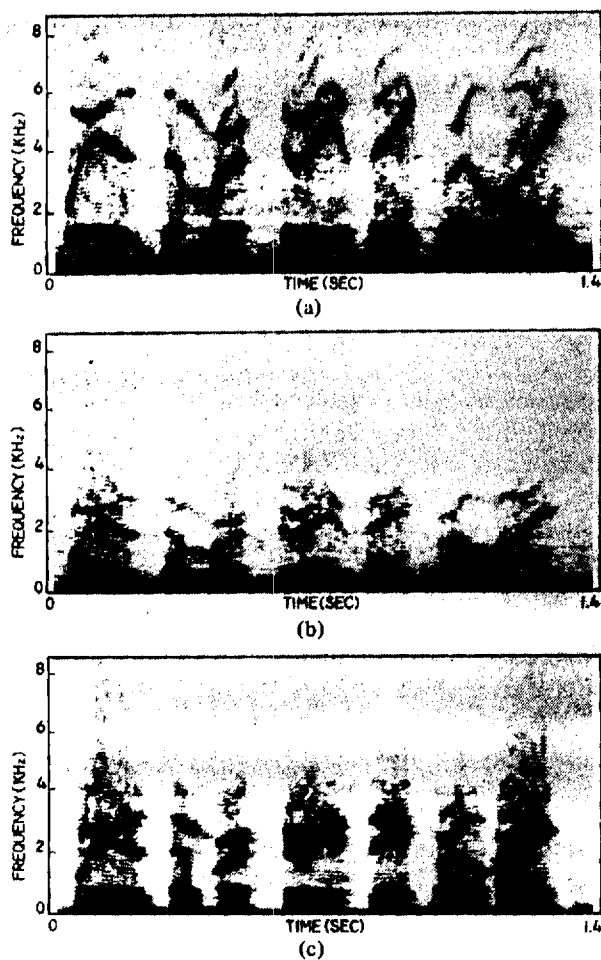


Fig. 11. Operation of the helium speech enhancement algorithm. (a) Spectrogram of helium speech utterance "where's the mercury vapor light?;" (b) spectrogram of enhanced speech with  $\alpha = 2$ ,  $F_0 = 0$ , piecewise-linear envelope estimation, and no noise suppression; (c) same as (b), but with noise suppression included. Note the noise burst at the end of the sentence and the speech suppression at 0.4 and 1.1 s.

formant frequency shift improved intelligibility more than the linear shift; whether noise reduction improved intelligibility; and whether the piecewise-linear envelope estimation method improved intelligibility more than the linear predictive method.

#### A. Test Procedures

Tape recordings of eight word lists spoken in a hyperbaric heliox atmosphere were made available by the Institute for Advanced Study of the Communication Process at the University of Florida. Particulars of the data collection methods are given in [10]. These recordings consisted of four 50 word lists from the Griffiths Rhyming Minimal Contrasts test [25] and are shown in Table I. The lists differ only in altering the initial or final consonant of each word. Each list was read by the same male speaker at depths of 560 feet and 1000 ft, with a different speaker for each list.

These raw data were collected into two "data classes," denoted the R560 and R1000 classes, according to depth. Six processed data classes were created from these two. These classes differ in the method of envelope estimation, the use of a linear or nonlinear formant shift, and the use of noise reduc-

TABLE II  
WORD LISTS USED IN THE INTELLIGIBILITY TESTS

List A	List B	List C	List D
1. SUD	SUM	SUB	SUN
2. FILL	FIG	FIN	FIZZ
3. BUST	JUST	RUST	GUST
4. HIP	RIP	TIP	DIP
5. LED	SHED	RED	WED
6. PEAK	PEAS	PEAL	PEACE
7. DONE	DUD	DUNG	DUB
8. TEN	PEN	DEN	HEN
9. DIG	DIN	DID	DIN
10. BAT	BATCH	BASH	BASS
11. TEETHE	TEER	TEASE	TEEL
12. WE'RE	WEAL	WEAVE	WEED
13. WILL	HILL	KILL	TILL
14. PICK	PIT	PIP	PIG
15. MAT	MAD	MATH	MAN
16. RENT	BENT	WENT	DENT
17. KICK	CHICK	THICK	PICK
18. TOP	HOP	POP	COP
19. BARK	DARK	MARK	LARK
20. ZIP	LIP	NIP	GYP
21. SAD	SAT	SAG	SACK
22. LAWS	LONG	LOG	LODGE
23. NEST	BEST	VEST	REST
24. PIN	SIN	TIN	WIN
25. TAB	TAN	TAM	TANG
26. THEE	DEE	LEE	KNEE
27. SHEEN	SHEAVE	SHEATHE	SHEATH
28. CUFF	CUB	CUT	CUP
29. LASH	LACK	LASS	LAUGH
30. SOLD	COLD	HOLD	TOLD
31. FEEL	REEL	SEAL	ZEAL
32. TOSS	TAJ	TONG	TALKS
33. SING	SIP	SIN	SIT
34. GALE	PALE	TALE	BALE
35. SHAME	GAME	CAME	SAME
36. PUP	PUFF	PUB	PUCK
37. DUMB	DUB	DUTH	DUFF
38. DIG	WIG	BIG	RIG
39. PASS	PATH	PACK	PAD
40. HATH	HASH	HALF	HAVE
41. PEEL	FEEL	EEL	HEEL
42. WIG	WITH	WIT	WITCH
43. VIE	THY	FIE	THIGH
44. FIN	TIN	SHIN	KIN
45. MAT	VAT	THAT	FAT
46. BEIGE	BASE	BAYED	BATHE
47. THIN	TIN	CHIN	SHIN
48. LEAVE	LIEGE	LEACH	LEASH
49. WAY	MAY	GAY	THEY
50. YORE	GORE	WORE	LORE

tion. The characteristics of the data classes are summarized in Table III. Because the raw data were unusually noise free, noise excerpted from the data of Fig. 11 was added to the N560 and NP560 data sets to create artificial, but realistic, noisy data.

Four audio tapes were created from these data sets. Each tape began with two unprocessed word lists from either the R560 or R1000 data set, followed by six processed word lists, two from one data class and four from another. Each of the four word lists (and thus each speaker) occurs at least once in each tape. Because the lists comprising each tape were not commingled, some training effects are possible in the scores.

Thirty-three native English speaking listeners were recruited from the Georgia Tech community; 82 percent were graduate students, while the rest were faculty and staff. Between seven and ten listeners auditioned each tape.

The tests were conducted using the facilities of the Georgia Tech Speech Quality Laboratory. A group of subjects listened to an audio tape over headphones. For each word they heard, they had a written list of the corresponding words from all four word lists before them, e.g., "sud," "sum," "sub," and "sun" for the first word in a list. The subjects would select which word they thought had been spoken, and enter their response ("1" for first, "2" for second, etc.) on a keypad device. The responses were automatically collected by a mini-computer system for later analysis. Approximately three

se  
re  
ar  
B.  
in  
the  
ov  
list  
list  
Ma  
scc  
hel  
dit  
cor  
T  
wis  
ing.  
poi  
fere  
So  
the  
abo  
cor  
PNI  
Mar

TABLE III  
CHARACTERISTICS OF THE INTELLIGIBILITY TEST DATA CLASSES

class	depth (feet)	bandwidth (KHz)	raw/processed	PWL/LP envelope estimate	Lin/Nonlin formant shift	noise added?	noise removed?
R560	560	8	R	—	—	—	—
R1000	1000	8	R	—	—	—	—
PNL560	560	4	P	PWL	NL	NO	NO
PNL1000	1000	4	P	PWL	NL	NO	NO
PL1000	1000	4	P	PWL	LIN	NO	NO
LP560	560	4	P	LP	NL	NO	NO
N560	560	4	P	PWL	NL	YES	NO
NP560	560	4	P	PWL	NL	YES	YES

TABLE IV  
INTELLIGIBILITY SCORES BY DATA CLASS

data class	average score (%) ( $\mu$ )	standard deviation (%) ( $\sigma$ )	# of scores averaged (n)
PNL560	70.9	8.0	32
PNL1000	69.0	8.3	34
LP560	67.9	6.7	32
PL1000	67.5	8.0	28
N560	61.5	7.6	40
NP560	58.0	9.7	24
R560	46.3 (37.8)	8.4 (7.5)	30
R1000	40.2 (35.0)	6.6 (7.7)	34
(M1000)	(72.5)	(8.3)	( $\approx$ 48)
(M560)	(71.9)	(11.4)	( $\approx$ 48)

seconds were allowed for a response, a fairly demanding requirement.

More details concerning the raw data, the data processing, and the test procedures are given in [5].

### B. Test Results

The intelligibility scores for the eight data classes are given in Table IV. The average scores for each data class are simply the total percentage of correctly identified words, averaged over all four speakers and over all listeners exposed to those lists. The extra data classes M1000 and M560 represent published scores [10] for the Marconi type O23 unscrambler. The Marconi device is of the general type described in [13]. The scores in parentheses for the R560 and R1000 data are the raw helium speech scores from the same study. Experimental conditions in that study differed from those here, so that direct comparisons may not be valid.

The significance of the system rankings can be checked pairwise using the one-tailed "studentized t-test." Generally speaking, a difference between scores of about three percentage points was found to be significant at the 0.05 level, while a difference of about five points is significant at the 0.01 level.

Several conclusions can now be drawn from Table IV. First, the STFT-based unscrambler improves intelligibility from about 40-45 percent to about 70 percent, as can be seen by comparing the R560 and R1000 data to the PNL560 and PNL1000 data. This performance appears equivalent to the Marconi device (M560, M1000), but comparison of these

scores is not conclusive (note the large difference in R560 and R1000 scores obtained here and in [10]).

The noise reduction algorithm can be judged using the N560 and NP560 scores. The N560 data show that untreated noise clearly reduces the intelligibility of the enhanced speech (compare to PNL560), but the noise-suppressed data NP560 show further reduction. The difference between the N560 and NP560 is marginally significant at the 0.05 level. Thus, although the noise reduction was thought to improve speech quality, it had a detrimental effect on the intelligibility.

The effect of the nonlinear term in the formant frequency shift is seen by comparing the PNL1000 and PL1000 scores. Although the nonlinear data score is higher, the difference is not significant at the 0.05 level. There is some reason to believe that this result understates the importance of the nonlinearity. This data base tests only consonant discrimination; vowel discrimination should be tested as well to better measure the importance of formant frequency differences. Also, phonemic analysis of the test results shows that the nonlinear shift increases recognition of noncontinuant phonemes with a significance of 0.01. Since formant transitions are important to recognition of these sounds, this supports the importance of a nonlinear shift. Nonetheless, the overall intelligibility scores on this data base do not conclusively establish the desirability of the nonlinear formant frequency shift.

Comparison of the linear predictive (LP560) and piecewise-linear (PNL560) scores shows that the piecewise-linear method gives higher intelligibility, with the difference marginally significant at the 0.05 level. This reinforces the preference developed earlier for the piecewise-linear method on computational and quality grounds.

It is interesting to note that, despite a large difference in unprocessed data scores at 560 and 1000 ft, the processor proposed here restores speech at both depths to nearly equal intelligibilities. This suggests that the enhanced speech intelligibility is nearly independent of depth over a wide range, once a depth is reached which reduces the processed speech intelligibility to about 70 percent. The data of [10] suggest that this depth is around 100 ft.

Phonemic confusion matrices were constructed for all data classes as well. The matrices, and a complete discussion of the results, are available in [5], but some results for the processed speech are worth mentioning here.

The beneficial effect of a nonlinear formant frequency shift on recognition of noncontinuants has already been noted. At

both depths, recognition of noncontinuants is *improved* more than for continuants, but recognition of continuants remains significantly better. One possible interpretation of these observations is that the window length is near the maximum value permissible before "smearing" of transient sounds becomes excessive, an explanation supported informally in [7]. More generally, the algorithm seems especially effective at reducing both the inter- and intra-class confusion among the nasals, stops, and glides, while it is less effective with the more noise-like fricatives and affricates. Finally, it is worthwhile to note that the unscrambler improves recognition of all 26 tested phonemes at 560 ft, and of 25 of them at 1000 ft.

### VIII. CONCLUSION

This paper has dealt with the modeling of helium speech, the design of a system for helium speech enhancement based on the short-time Fourier transform, and the evaluation of that system. In regard to helium speech modeling, it has been seen that the formant bandwidths will increase greatly in hyperbaric heliox, contrary to some previous statements, and that the behavior of the relative formant amplitudes is not well understood. Improved modeling of helium speech phenomena is likely the greatest need if significant improvements in unscramblers are to be realized. To this end, it would be of great interest to construct a complete vocal cord-vocal tract simulation as in [8] and exercise it under normal and hyperbaric heliox conditions. Such a simulation should provide new clues to formant, glottal, and pitch behavior.

The proposed unscrambler is based on a model for the short-time Fourier transform of speech. The key step is estimation of the spectral envelope from the STFT, and a simple "piecewise-linear" algorithm has been developed for this purpose. The unscrambler requires neither pitch nor voicing decisions; this gives it a robustness which may be lacking in other devices, such as the Marconi unit [26]. It can achieve arbitrary formant frequency mappings and can halve the signal bandwidth prior to output synthesis. Its chief disadvantage is its relatively high computational requirements.

Formal subjective intelligibility tests have been used to evaluate the unscrambler. Intelligibility is increased from 40-45 percent to about 70 percent. This is believed to be about equivalent to the simpler Marconi unscrambler. An attempt to determine the importance of the nonlinear term in the formant frequency shift was inconclusive. Evaluation on a broader data base which tested vowel discrimination would be helpful here, since this is one of the major rationales for more complex processors such as the ones proposed here and in [4]. The noise reduction algorithm was found to be detrimental to intelligibility. Clearly, more work is needed in noise spectrum estimation and speech/noise discrimination in high-noise environments. Frequency-dependent subtraction might also ease the speech distortion problems caused by non-white noise.

There is no guarantee that the needed improvements suggested above would lead to more than marginal improvements in the effectiveness of the unscrambler. Good deep-submergence communication demands a systems approach. The environment must be kept as quiet as feasible so as to attack the noise problem at its source. Microphones with good

noise-cancelling characteristics and a high bandwidth, density-independent frequency response are needed. Unscramblers must be made as sophisticated as possible. In practice, the use of trained divers will improve results over those derived from naive listeners. It is the combination of all of these improvements which will lead to effective deep-sea speech communications.

### ACKNOWLEDGMENTS

The author would like to thank Dr. R. W. Schafer for introducing him to the topics discussed in this paper, and for many discussions and suggestions on all aspects of this work; and Dr. T. P. Barnwell, III, for his assistance with the intelligibility testing and analysis.

### REFERENCES

- [1] C. T. Morrow, "Speech in deep-submergence atmospheres," *J. Acoust. Soc. Amer.*, vol. 50, no. 3, pp. 715-728, 1971.
- [2] T. A. Giordano, H. B. Rothman, and H. Hollien, "Helium speech unscramblers—A critical review of the state of the art," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 436-444, Oct. 1973.
- [3] G. Fant and J. Lindquist, "Pressure and gas mixture effects on diver's speech," Speech Transmission Laboratory, Royal Inst. of Tech., Stockholm, Sweden, Quarterly Progress and Status Rep. STL-QPSR-1/1968, pp. 7-17, 1968.
- [4] R. F. Quick, Jr., "Helium speech translation using homomorphic techniques," Air Force Cambridge Research Laboratories, Rep. AFCRL-70-0424, July 1970.
- [5] M. A. Richards, "Helium speech enhancement using the short-time Fourier transform," Ph.D. dissertation, School of Electrical Engineering, Georgia Institute of Technology, Mar. 1982.
- [6] J. L. Flanagan, *Speech Analysis, Synthesis, and Perception*, 2nd ed. New York: Springer-Verlag, 1972.
- [7] M. Vestrheim, S. Hatlestad, E. Belcher, and K. Slethei, "Deep ex-81 diver communication," Norwegian Underwater Technology Center, Rep. 17-82, Jan. 1982.
- [8] J. L. Flanagan, K. Ishizaka, and K. L. Shipley, "Synthesis of speech from a dynamic model of the vocal cords and vocal tract," *Bell Syst. Tech. J.*, vol. 54, no. 3, pp. 485-506, Mar. 1975.
- [9] M. Nakatsui, "Comments on helium speech—Insight into speech event needed," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 472-473, Dec. 1974.
- [10] H. B. Rothman, R. Gelfand, H. Hollien, and C. J. Lambertsen, "Speech intelligibility at high helium-oxygen pressures," *Undersea Biomed. Res.*, vol. 7, no. 4, pp. 265-275, Dec. 1980.
- [11] D. J. MacLean, "Analysis of speech in a helium-oxygen mixture under pressure," *J. Acoust. Soc. Amer.*, vol. 40, no. 3, pp. 625-627, 1966.
- [12] J. Suzuki and M. Nakatsui, "Translation of helium speech by splicing of autocorrelation function," *J. Radio Res. Lab.*, Japan, vol. 23, no. 111, pp. 229-234, July 1976.
- [13] W. R. Stover, "Technique for correcting helium speech distortion," *J. Acoust. Soc. Amer.*, vol. 41, no. 1, pp. 70-74, 1967.
- [14] H. Suzuki and G. Ooyama, "Helium speech unscrambler using a digital filter constructed by linear prediction and impulse response conversion," *Electr. Commun. Japan*, vol. 58-A, no. 6, pp. 68-75, 1975.
- [15] R. M. Golden, "Improving naturalness and intelligibility of helium-oxygen speech, using vocoder techniques," *J. Acoust. Soc. Amer.*, vol. 40, no. 3, pp. 621-624, 1966.
- [16] T. Takasugi and J. Suzuki, "Translation of helium speech by the use of 'analytic signal,'" *J. Radio Res. Lab.*, Japan, vol. 21, no. 103, pp. 61-69, 1974.
- [17] M. Copel, "Helium voice unscrambling," *IEEE Trans. Audio Electroacoust.*, vol. AU-14, pp. 122-126, Sept. 1966.
- [18] M. R. Portnoff, "Time-frequency representation of digital signals and systems based on short-time Fourier analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 55-69, Feb. 1980.
- [19] —, "Short-time Fourier analysis of sampled speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29 (part I), pp. 364-373, June 1981.

- [20] M. Berouti, R. Schwartz, and J. Makhoul, "Enhancement of speech corrupted by acoustic noise," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pp. 208-211, Apr. 1979.
- [21] J. Makhoul, "Methods for nonlinear spectral distortion of speech signals," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pp. 87-90, Apr. 1976.
- [22] A. V. Oppenheim and R. W. Schaffer, *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.
- [23] D. B. Paul, "The spectral envelope estimation vocoder," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 786-794, Aug. 1981.
- [24] L. R. Rabiner and R. W. Schaffer, *Digital Processing of Speech Signals*. Englewood Cliffs, NJ: Prentice-Hall, 1978.
- [25] J. D. Griffiths, "Rhyming minimal contrasts: A simplified diagnostic articulation test," *J. Acoust. Soc. Amer.*, vol. 42, no. 1, pp. 236-241, 1967.
- [26] D. D. Brown and S. H. Feinstein, "An evaluation of three helium speech unscramblers to a depth of 1000 feet," *J. Sound and Vibration*, vol. 48, no. 1, pp. 123-135, 1976.



Mark A. Richards received the B.E.E. degree with honor from the Georgia Institute of Technology, Atlanta, in 1974, the M.S.E.E. degree from Stanford University, Stanford, CA, in 1976, and the Ph.D. degree in electrical engineering from Georgia Tech in 1982. His thesis dealt with enhancement of helium speech using short-time Fourier analysis techniques.

He was employed as a summer intern at the NASA Manned Spacecraft Center in Houston, TX, in 1974. From 1975 to 1976 he was with

ESL, Inc., Sunnyvale, CA. While in the graduate program at Georgia Tech, he served as a Research or Teaching Assistant at various times. He is now a Research Engineer with the Radar and Instrumentation Laboratory of the Georgia Tech Engineering Experiment Station. His current interests include speech enhancement, radar target identification, and time-frequency analysis of signals.

Dr. Richards is a member of the Audio Engineering Society, Eta Kappa Nu, and Sigma Xi.

# A Conversational Test for Comparing Voice Systems Using Working Two-Way Communication Links

ASTRID SCHMIDT-NIELSEN AND STEPHANIE S. EVERETT

**Abstract**—A conversational test using live two-way communications provides a measure of the actual usability of voice systems, especially when voice quality is degraded. A conversational test developed at NRL was compared with two other communicability tests in a series of experiments using a variety of digital voice processors with data rates from 800 to 32 000 bits/s. All three tests ranked the voice processors very similarly, but they did not discriminate equally well among different processors. Other advantages and disadvantages of conversational test methods are discussed.

## I. INTRODUCTION

VOICE testing of communication equipment and systems serves three primary purposes: selection, evaluation, and development. When there is a choice of several competing systems (e.g., different manufacturers) tests are needed to determine which one has the best quality and intelligibility. When any changes or "improvements" are made, tests are used to decide whether intelligibility is actually better or worse. Finally, tests of existing systems can be used to determine weaknesses and to decide how future systems might be improved.

Most voice tests can be grouped into three major types, although a few may bridge these categories:

*Intelligibility or phoneme tests* assess the ability to hear or discriminate among individual speech sounds. The test materials are usually words or syllables, and the score is based on the number of correct discriminations or identifications.

*Quality or rating tests* are used to obtain opinion measures

and assess acceptability rather than intelligibility per se. The test materials consist of one or more sentences which are rated by the listeners on various rating scales.

*Conversational or communicability tests* assess the usability of a system using a two-way communication task. This allows the users to interact and to adapt to the requirements of the system (e.g., talk louder, talk more slowly, ask for repeats, etc.). On completion of the task the usability of the system is rated on one or more scales.

Several excellent measures of intelligibility and quality are available, e.g., modified rhyme test (MRT) (House, Williams, Hecker, and Kryter [1]), diagnostic rhyme test (DRT) (Voiers [2]), diagnostic acceptability measure (DAM) (Voiers [3]).

In situations where voice quality can be expected to be seriously degraded, as for example in low data rate and very low data rate digital voice communications, it becomes increasingly important to evaluate the communicability or actual usability of the voice system in addition to obtaining intelligibility scores. Only an interactive test can be used to distinguish between degradations that can be overcome by learning compensatory behaviors and those that can not. A real time measure such as a conversational test is also necessary to evaluate degradation due to real time effects such as transmission or processing delays.

## II. BACKGROUND

The general format of a conversational test consists of a communication task requiring an exchange of information between the participants, followed by an evaluation of the ease or difficulty of using the voice system.

Manuscript received November 24, 1981; revised May 26, 1982. This work was supported by ONR 6.1 funds (RRO21-05-42).

The authors are with the Naval Research Laboratory, Washington, DC 20375.

general Cooley-Tukey algorithms in [2]. This paper should logically have preceded [2], since the DFT algorithm which is derived in this paper is evaluated using the generalized Cooley-Tukey algorithm in [2]. This current paper and [2] taken together are a generalization of the algorithms for hexagonal sampling which are presented in [1].

This paper will begin by discussing multidimensional sampling itself. While most of this material appears in the key paper by Petersen and Middleton [3], a review of the major results is unavoidable since the sampling operation plays such a central role in what is to follow. This review will also allow us to establish a matrix notation that will not only simplify our mathematical expressions, but will also make the analogy with the 1- $D$  case clearer. The following section will concern linear shift-invariant systems and will discuss convolution, difference equations, and Fourier transforms. Section IV will discuss generalized discrete Fourier transforms and discrete spectral analysis. Finally, in the last section we will discuss the problems of decimation and interpolation on multidimensional sampling lattices. By using general decimators and interpolators we can solve the problem of interpolating from one lattice (e.g., a 3- $D$  body-centered cubic lattice) to another (e.g., a 3- $D$  cubic lattice). The approach to this problem discussed in this paper represents a better solution to this problem than the algorithm discussed in [4].

## II. MULTIDIMENSIONAL PERIODIC SAMPLING

One-dimensional periodic sampling of a band-limited signal  $x_a(t)$  corresponds to forming a sequence of numbers by evaluating  $x_a(t)$  at equispaced values of its argument. Thus, if  $x(n)$  is used to denote the sequence of values, we form

$$x(n) = x_a(nT) \quad (1)$$

for all integer values of  $n$ . (The subscript "a" in (1) denotes the fact that  $x_a(t)$  is an analog waveform.) The parameter  $T$  is called the *sampling period* and  $1/T$  is called the *sampling rate*. In extending this concept to permit the sampling of  $D$ -dimensional signals, the set of equispaced sample locations becomes a  $D$ -dimensional lattice.

A  $D$ -dimensional lattice is formed by taking all integer linear combinations of a set of  $D$  linearly independent (column) vectors  $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_D\}$ . An example of a 2- $D$  lattice is shown in Fig. 1. Collectively, these vectors are known as the *basis* of the lattice. The *sampling matrix*,  $\mathbf{V}$ , is a  $D \times D$  matrix whose columns form the basis. Thus,

$$\mathbf{V} = [\mathbf{v}_1 \mid \mathbf{v}_2 \mid \dots \mid \mathbf{v}_D]. \quad (2)$$

Each location in the sampling lattice can then be expressed as

$$\mathbf{t} = \mathbf{V}\mathbf{n} \quad (3)$$

where  $\mathbf{n}$  is a vector with integer entries. The most common sampling lattice is the rectangular one, for which  $\mathbf{V}$  is diagonal.

For a given lattice neither the basis nor the sampling matrix is unique. If  $\mathbf{E}$  is a  $D \times D$  matrix of integers such that  $|\det \mathbf{E}| = 1$  (such a matrix is called *unimodular*), then  $\hat{\mathbf{V}}$ , given by

$$\hat{\mathbf{V}} = \mathbf{E}\mathbf{V} \quad (4)$$

and  $\mathbf{V}$  define the same lattice. The quantity  $|\det \mathbf{V}|$ , however,

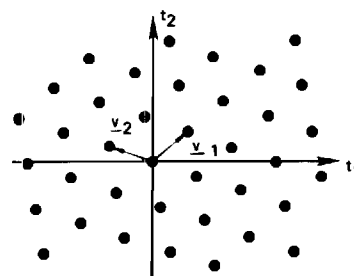


Fig. 1. A two-dimensional sampling lattice.

is unique for a given lattice and it physically corresponds to the reciprocal of the sampling density (sampling rate) [4].

If  $x_a(\mathbf{t})$  is sampled on the lattice which is defined by (3), the sequence of samples  $x(\mathbf{n})$  is given by

$$x(\mathbf{n}) = x_a(\mathbf{V}\mathbf{n}) \quad (5)$$

which bears an obvious resemblance to (1).

Recall that in the 1- $D$  case, if  $x(n) = x_a(nT)$ , then

$$X(\omega) = \frac{1}{T} \sum_{r=-\infty}^{\infty} X_a\left(\omega - \frac{2\pi r}{T}\right) \quad (6)$$

which says that the spectrum of the sequence is an aliased version of the spectrum of  $x_a(t)$ . If  $X_a(\omega) \equiv 0$  for  $|\omega| \geq \pi/T$ ,  $X_a(\omega)$  can be recovered exactly from  $X(\omega)$ . An analogous situation occurs in the  $D$ -dimensional case. Here, however, the degree of aliasing depends not only on the sampling density, but also upon the geometry of the sampling lattice.

Let the  $D$ -dimensional Fourier transform of  $x_a(\mathbf{t})$  be defined as

$$X_a(\boldsymbol{\omega}) = \int_{-\infty}^{\infty} x_a(\mathbf{t}) \exp[-j\boldsymbol{\omega}^T \mathbf{t}] d\mathbf{t} \quad (7)$$

where  $\boldsymbol{\omega}^T$  denotes the transpose of  $\boldsymbol{\omega}$ , and where the integral is evaluated over all of  $\mathbf{t}$ -space. Similarly, let the Fourier transform of the sequence  $x(\mathbf{n})$  be defined as

$$X(\boldsymbol{\omega}) = \sum_{\mathbf{n}=-\infty}^{\infty} x(\mathbf{n}) \exp[-j\boldsymbol{\omega}^T \mathbf{V}\mathbf{n}]. \quad (8)$$

Then, if  $x(\mathbf{n}) = x_a(\mathbf{V}\mathbf{n})$ , it can be shown using techniques enumerated in [3] that

$$X(\boldsymbol{\omega}) = \frac{1}{|\det \mathbf{V}|} \sum_{r=-\infty}^{\infty} X_a(\boldsymbol{\omega} - \mathbf{U}\mathbf{r}) \quad (9)$$

where

$$\mathbf{U}^T \mathbf{V} = 2\pi \mathbf{I} \quad (10)$$

and  $\mathbf{I}$  is the  $D \times D$  identity matrix. An aliased spectrum is depicted in Fig. 2. In the area of mathematics known as the geometry of numbers, the lattice formed by  $\mathbf{U}$  is known as the *polar lattice* [4]. It is also known as the *reciprocal lattice*, a term we prefer. The matrix  $\mathbf{U}$  will be called the *aliasing matrix*.

$X_a(\boldsymbol{\omega})$  is *band-limited* with a spectrum limited to a region  $\mathcal{W}$  of Fourier space if

$$X_a(\boldsymbol{\omega}) = 0 \quad \boldsymbol{\omega} \notin \mathcal{W}. \quad (11)$$

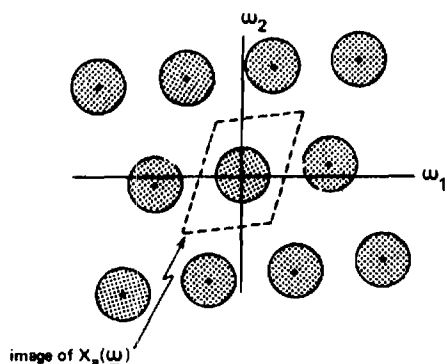


Fig. 2. An aliased spectrum resulting from the sampling of a band-limited signal on a periodic lattice.

Band-limited signals can sometimes be recovered exactly from their sample values. Such an exact recovery is possible if the functions  $X_a(\boldsymbol{\omega} - \mathbf{U}\mathbf{r})$ , for all integer vectors  $\mathbf{r}$ , do not possess overlapping regions of support. Whether or not this condition is met depends both upon the shape of  $W$  and on the geometry of the reciprocal lattice. If, however, this condition is met, then

$$X_a(\boldsymbol{\omega}) = \begin{cases} |\det \mathbf{V}| X(\boldsymbol{\omega}), & \boldsymbol{\omega} \in W \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

and

$$x_a(\mathbf{t}) = \sum_{\mathbf{n}} x_a(\mathbf{V}\mathbf{n}) \phi(\mathbf{t} - \mathbf{V}\mathbf{n}) \quad (13)$$

$$\phi(\mathbf{t}) = \frac{1}{(2\pi)^D} \int_W \exp[-j\boldsymbol{\omega}^T \mathbf{t}] d\boldsymbol{\omega}. \quad (14)$$

Two special 2- $D$  cases are worthy of mention at this point. The first is rectangular sampling, for which

$$\mathbf{V}_R = \begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix} \quad (15)$$

and

$$\mathbf{U}_R = \begin{bmatrix} \frac{2\pi}{T_1} & 0 \\ 0 & \frac{2\pi}{T_2} \end{bmatrix}. \quad (16)$$

In this case both the sampling and reciprocal lattices are rectangular. The second special case corresponds to hexagonal sampling, for which

$$\mathbf{V}_H = \begin{bmatrix} \frac{T_1}{2} & \frac{T_1}{2} \\ T_2 & -T_2 \end{bmatrix} \quad (17)$$

$$\mathbf{U}_H = \begin{bmatrix} \frac{2\pi}{T_1} & \frac{2\pi}{T_1} \\ \frac{\pi}{T_2} & -\frac{\pi}{T_2} \end{bmatrix}. \quad (18)$$

### III. LINEAR SHIFT-INVARIANT SYSTEMS

Consider a  $D$ -dimensional linear shift-invariant system with input array  $\mathbf{x}(\mathbf{n})$  and output array  $\mathbf{y}(\mathbf{n})$ , where both arrays are defined over identical lattices. The input and output arrays are then related by the convolution sum

$$\mathbf{y}(\mathbf{n}) = \sum_{\mathbf{k}} \mathbf{x}(\mathbf{k}) h(\mathbf{n} - \mathbf{k}) \quad (19)$$

where  $h(\mathbf{n})$ , the *impulse response*, is the response of the system to the signal

$$\delta(\mathbf{n}) = \begin{cases} 1, & \mathbf{n} = \mathbf{0} \\ 0, & \mathbf{n} \neq \mathbf{0} \end{cases} \quad (20)$$

It should be observed that the form of the convolution sum is independent of the sampling matrix  $\mathbf{V}$ .

#### A. Frequency Response

If we let the input to a linear shift-invariant system be a sampled complex sinusoid of the form

$$\mathbf{x}(\mathbf{n}) = \exp[j\boldsymbol{\omega}^T \mathbf{V}\mathbf{n}] \quad (21)$$

then, using (19), the system output is seen to be

$$\mathbf{y}(\mathbf{n}) = \exp[j\boldsymbol{\omega}^T \mathbf{V}\mathbf{n}] \sum_{\mathbf{k}} h(\mathbf{k}) \exp[-j\boldsymbol{\omega}^T \mathbf{V}\mathbf{k}]. \quad (22)$$

Sampled complex sinusoids of the form of (20) are thus eigenfunctions of periodically sampled linear shift-invariant systems. This leads us to define the *frequency response* of an LSI system as the corresponding eigenvalue

$$H(\boldsymbol{\omega}) = \sum_{\mathbf{n}} h(\mathbf{n}) \exp[-j\boldsymbol{\omega}^T \mathbf{V}\mathbf{n}]. \quad (23)$$

The frequency response  $H(\boldsymbol{\omega})$  is a periodic function of  $\boldsymbol{\omega}$  with periodicity matrix  $\mathbf{U}$ , where  $\mathbf{U}^T \mathbf{V} = 2\pi \mathbf{I}$ . By this we mean that

$$H(\boldsymbol{\omega}) = H(\boldsymbol{\omega} + \mathbf{U}\mathbf{r}) \quad (24)$$

for any integer vector  $\mathbf{r}$ .

Let  $I_H$  denote any period of  $H(\boldsymbol{\omega})$ . This period has a volume of  $|\det \mathbf{U}|$ . The frequency response can then be inverted by performing the integral

$$h(\mathbf{n}) = \frac{1}{|\det \mathbf{U}|} \int_{I_H} H(\boldsymbol{\omega}) \exp[j\boldsymbol{\omega}^T \mathbf{V}\mathbf{n}] d\boldsymbol{\omega}. \quad (25)$$

To verify the validity of this relation we can substitute (23) into (25) and exploit the periodicity of the exponentials.

#### B. The Fourier Transform

We have already seen that if the Fourier transform of a sequence  $\mathbf{x}(\mathbf{n})$  is defined as

$$X(\boldsymbol{\omega}) = \sum_{\mathbf{n}=-\infty}^{\infty} \mathbf{x}(\mathbf{n}) \exp[-j\boldsymbol{\omega}^T \mathbf{V}\mathbf{n}] \quad (26)$$

then the Fourier transform of a sequence and the Fourier transform of the band-limited signal from which it is derived

are simply related by aliasing. This definition is consistent with the one-dimensional one. Since the frequency response is the Fourier transform of the impulse response, like the frequency response, the Fourier transform  $X(\boldsymbol{\omega})$  is periodic with periodicity matrix  $\mathbf{U} = 2\pi(\mathbf{V}^T)^{-1}$  and it can be inverted by using (25).

The general Fourier transform has properties which are identical to the corresponding properties of 1-D Fourier transforms, except for the fine details of the resulting expression. In particular, the transform is linear and the convolution theorem holds. Thus, if  $x(\mathbf{n})$  is the input to a linear shift-invariant system and  $y(\mathbf{n})$  is the corresponding output, we can write

$$Y(\boldsymbol{\omega}) = X(\boldsymbol{\omega})H(\boldsymbol{\omega}). \quad (27)$$

Two Fourier transform properties which look slightly different are the shift-property and Parseval's relation. The former states that

$$F[x(\mathbf{n} - \mathbf{n}_0)] = \exp[-j\boldsymbol{\omega}^T \mathbf{V}\mathbf{n}_0] X(\boldsymbol{\omega}) \quad (28)$$

where  $F[\cdot]$  denotes the Fourier transform operator. Parseval's relation can be written as

$$\sum_{\mathbf{n}} x(\mathbf{n})y^*(\mathbf{n}) = \frac{1}{|\det \mathbf{U}|} \int_{J_{\mathbf{V}}} X(\boldsymbol{\omega}) Y^*(\boldsymbol{\omega}) d\boldsymbol{\omega}. \quad (29)$$

where  $J_{\mathbf{V}}$  is one period of  $X(\boldsymbol{\omega}) Y^*(\boldsymbol{\omega})$ .

### C. Difference Equations

A periodically sampled LSI system can be defined implicitly by means of a linear constant coefficient difference equation of the form

$$\sum_{\mathbf{k}} b(\mathbf{k})y(\mathbf{n} - \mathbf{k}) = \sum_{\mathbf{k}} a(\mathbf{k})x(\mathbf{n} - \mathbf{k}). \quad (30)$$

where the sums each involve only a finite number of terms. Since the form of this difference equation is completely independent of the sampling matrix  $\mathbf{V}$ , the same hardware or software realization of the filter can be used whether the input and output arrays are rectangularly sampled, hexagonally sampled, or whether a more general sampling matrix is used. The stability of the difference equation is also independent of  $\mathbf{V}$ .

## IV. DISCRETE FOURIER TRANSFORMS (DFT)

The DFT is an exact Fourier representation for periodically sampled arrays with a finite number of nonzero samples. It assumes the form of a periodically sampled Fourier transform. As in the 1-D case, the DFT can be interpreted as a Fourier series representation for one period of a periodic sequence. It is thus easiest if we begin with a discussion of periodic sequences.

A sequence  $\tilde{x}(\mathbf{n})$  is periodic if it satisfies the relation

$$\tilde{x}(\mathbf{n} + \mathbf{N}\mathbf{r}) = \tilde{x}(\mathbf{n}) \quad (31)$$

$\det \mathbf{N} \neq 0$

for all integer vectors  $\mathbf{n}$  and  $\mathbf{r}$ . The integer matrix  $\mathbf{N}$  is called

the *periodicity matrix* and the number of samples in one period of the signal is given by  $|\det \mathbf{N}|$ . For a given periodic sequence the periodicity matrix is not unique; it can be multiplied by any unimodular integer matrix and still describe the same periodic signal. These facts follow by analogy from the corresponding facts concerning sampling matrices.

The role of the periodicity matrix can be explained somewhat by reference to Fig. 3. There we see three different periodic signals each of which has as its fundamental period an  $N_1 \times N_2$  rectangular block of samples. What distinguishes these three signals is the manner in which these blocks are abutted together. The columns of the periodicity matrix are vectors which denote the displacement from a sample on one period to the corresponding sample on another period. Thus, for the rectangular arrangement in Fig. 3(a) an acceptable periodicity matrix is

$$\mathbf{N} = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (32)$$

For the hexagonal arrangement in Fig. 3(b) one possible periodicity matrix is

$$\mathbf{N} = \begin{bmatrix} N_1 & N_2/2 \\ N_1 & -N_2/2 \end{bmatrix} \quad (33)$$

(where  $N_2$  is assumed to be divisible by 2) and for the final arrangement, a periodicity matrix is given by

$$\mathbf{N} = \begin{bmatrix} N_1 & -1 \\ 0 & N_2 \end{bmatrix}. \quad (34)$$

Any periodic sequence  $\tilde{x}(\mathbf{n})$  can be represented as a sum of harmonically related complex exponentials. To see this we can first take the Fourier transform of both sides of (31) using the shift property (28). This yields

$$\tilde{X}(\boldsymbol{\omega}) = \exp[-j\boldsymbol{\omega}^T \mathbf{V}\mathbf{N}\mathbf{r}] \tilde{X}(\boldsymbol{\omega}). \quad (35)$$

Thus, for all frequencies at which  $\tilde{X}(\boldsymbol{\omega}) \neq 0$  we have

$$\exp[-j\boldsymbol{\omega}^T \mathbf{V}\mathbf{N}\mathbf{r}] = 1 \quad (36)$$

or

$$\boldsymbol{\omega}^T = 2\pi\mathbf{k}^T(\mathbf{V}\mathbf{N})^{-1} \quad (37)$$

where  $\mathbf{k}$  is a vector of integers. This says that the only frequency components that a periodic array can have are harmonically related. Thus, we can write

$$\tilde{x}(\mathbf{n}) = \sum_{\mathbf{k} \in J_{\mathbf{N}}} \tilde{X}(\mathbf{k}) \exp[j(2\pi\mathbf{k}^T(\mathbf{V}\mathbf{N})^{-1} \mathbf{V}\mathbf{n})] \quad (38)$$

$$= \sum_{\mathbf{k} \in J_{\mathbf{N}}} \tilde{X}(\mathbf{k}) \exp[j(2\pi\mathbf{k}^T \mathbf{N}^{-1} \mathbf{n})]. \quad (39)$$

The exponential  $\exp[j(2\pi\mathbf{k}^T \mathbf{N}^{-1} \mathbf{n})]$  is periodic in  $\mathbf{n}$  with periodicity matrix  $\mathbf{N}$  and is periodic in  $\mathbf{k}$  with periodicity matrix  $\mathbf{N}^T$ . As a consequence, these exponentials are distinct for only  $|\det \mathbf{N}|$  values of  $\mathbf{k}$ . It is sufficient, therefore, to limit the sum to a region  $J_{\mathbf{N}}$  which contains one period of  $\exp[j(2\pi\mathbf{k}^T \mathbf{N}^{-1} \mathbf{n})]$  considered as a function of  $\mathbf{k}$ .

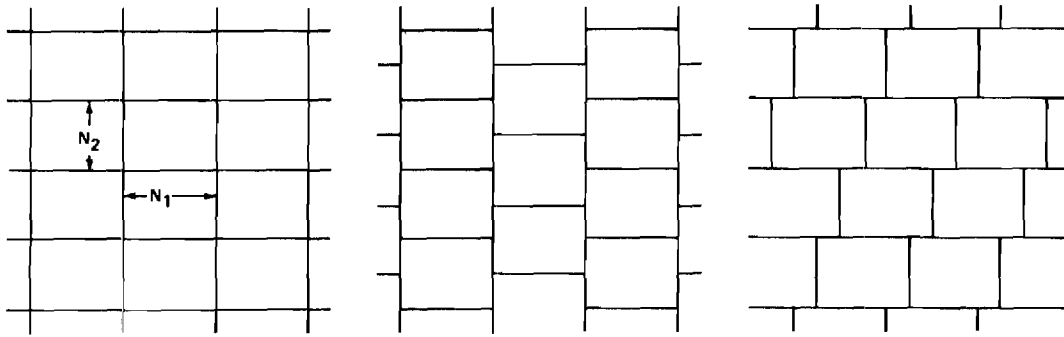


Fig. 3. Three different periodic signals which have the same fundamental period.

The Fourier series coefficients  $\tilde{X}(\mathbf{k})$  can be evaluated using the relation

$$\tilde{X}(\mathbf{k}) = \frac{1}{|\det \mathbf{N}|} \sum_{\mathbf{n} \in I_{\mathbf{N}}} \tilde{x}(\mathbf{n}) \exp[-j\mathbf{k}^T(2\pi\mathbf{N}^{-1})\mathbf{n}] \quad (40)$$

where  $I_{\mathbf{N}}$  denotes one period of  $\tilde{x}(\mathbf{n})$ .

If we now consider  $x(\mathbf{n})$  to be a sequence with finite support limited to  $I_{\mathbf{N}}$ ,  $\tilde{x}(\mathbf{n})$  can be considered to be its periodic extension. By varying  $\mathbf{N}$  we vary the manner by which the periodic extension is formed. This results in the discrete Fourier transform for periodically sampled signals.

$$X(\mathbf{k}) = \sum_{\mathbf{n} \in I_{\mathbf{N}}} x(\mathbf{n}) \exp[-j\mathbf{k}^T(2\pi\mathbf{N}^{-1})\mathbf{n}] \quad (41)$$

$$x(\mathbf{n}) = \frac{1}{|\det \mathbf{N}|} \sum_{\mathbf{k} \in J_{\mathbf{N}}} X(\mathbf{k}) \exp[j\mathbf{k}^T(2\pi\mathbf{N}^{-1})\mathbf{n}]. \quad (42)$$

In keeping with standard usage we have moved the normalization constant from the forward to the inverse transform. It should be noted that this reduces to the normal DFT in the 1-D case and to the familiar rectangular multidimensional DFT when  $\mathbf{N}$  is diagonal.

The numbers  $X(\mathbf{k})$  can be interpreted as samples of the Fourier transform of  $x(\mathbf{n})$ . We can see this by comparing (26) with (41). The DFT corresponds to evaluating the Fourier transform at these values of  $\boldsymbol{\omega}$  for which

$$\mathbf{k}^T(2\pi\mathbf{N}^{-1}) = \boldsymbol{\omega}^T \mathbf{V}$$

or

$$\begin{aligned} \boldsymbol{\omega} &= (\mathbf{V}^{-1})^T(2\pi\mathbf{N}^{-1})^T \mathbf{k} \\ &= \mathbf{U}(\mathbf{N}^{-1})^T \mathbf{k}. \end{aligned} \quad (43)$$

Thus, the matrix  $\mathbf{R} = \mathbf{U}(\mathbf{N}^{-1})^T$  serves as a Fourier domain sampling matrix. This provides one method for choosing the periodicity matrix  $\mathbf{N}$ . It must contain integer entries and it must be consistent with the region of support of  $x(\mathbf{n})$ . Apart from these conditions,  $\mathbf{N}$  can be chosen to place the DFT samples where desired.

As an illustration of this fact, we might consider the three periodicity matrices

$$\mathbf{N}_1 = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (44)$$

$$\mathbf{N}_2 = \begin{bmatrix} N_1 & N_1 \\ N_2 & -N_2 \end{bmatrix} \quad (45)$$

$$\mathbf{N}_3 = \begin{bmatrix} N_1 & N_2 \\ N_2 & N_1 \end{bmatrix}. \quad (46)$$

$\mathbf{N}_1$  transforms a rectangularly sampled  $N_1 \times N_2$  2-D array into rectangular samples of its Fourier transform. It thus represents the traditional DFT.  $\mathbf{N}_3$  transforms a hexagonally sampled sequence into hexagonal samples of its Fourier transform. This transform was discussed extensively in [1]. The middle periodicity matrix  $\mathbf{N}_2$  is a hybrid. It transforms a rectangularly sampled sequence into hexagonal samples of its Fourier transform.

Almost any algorithm for evaluating the 1-D DFT can be generalized to permit the evaluation of the DFT in (41). In [1] the Cooley-Tukey algorithm is generalized. The Cooley-Tukey algorithm exploits structure which is present in the 1-D DFT when  $N$ , the length of the 1-D DFT is a highly composite integer. A similar exploitable structure exists in the multidimensional case when  $\mathbf{N}$ , the periodicity matrix can be factored into nontrivial integer matrix factors.

## V. DECIMATION AND INTERPOLATION

The problems of decimation and interpolation for periodically sampled multidimensional sequences are similar in some respects to their one-dimensional counterparts [6], but their description is hampered by the fact that multidimensional bandwidths have shapes as well as size and by our desire to be completely general. In addition, in the multidimensional case, decimators and interpolators affect not only the sampling rate, but also the geometry of the sampling lattice. We will demonstrate this at the end of this section with a specific example which converts from rectangular to hexagonal samples.

We can begin by considering the problem of decimation. Let  $x_a(\mathbf{t})$  be a band-limited analog signal which has been sampled using the sampling matrix  $\mathbf{V}$  to produce the sequence  $x(\mathbf{n}) = x_a(\mathbf{V}\mathbf{n})$ . Let  $I_X$  denote the region of support of the band-limited signal and let  $I_V$  denote one period of the Fourier transform of the sequence. If the conditions of the sampling theorem have been satisfied, then  $I_X \subset I_V$  and

$$X(\boldsymbol{\omega}) = \frac{1}{|\det \mathbf{V}|} \sum_{r=-\infty}^{\infty} X_a(\boldsymbol{\omega} - \mathbf{U}\mathbf{r}). \quad (47)$$

Now let  $y(\mathbf{m}) = x(\mathbf{D}\mathbf{m})$  be defined, which is a downsampled version of  $x(\mathbf{n})$ . The *downsampling matrix*  $\mathbf{D}$  is an integer matrix with  $\det \mathbf{D} \neq 0$ . Since we can also write  $y(\mathbf{m}) = x_a(\mathbf{V}\mathbf{D}\mathbf{m})$ , relating  $y(\mathbf{m})$  to the original analog signal  $x_a(\mathbf{t})$ , it follows that

$$Y(\boldsymbol{\omega}) = \frac{1}{|\det \mathbf{V}| \cdot |\det \mathbf{D}|} \sum_{r=-\infty}^{\infty} X_a(\boldsymbol{\omega} - (\mathbf{D}^T)^{-1}\mathbf{U}\mathbf{r}). \quad (48)$$

If  $\mathbf{D}$  is unimodular, then  $\mathbf{V}$  and  $\mathbf{V}\mathbf{D}$  define the same sampling

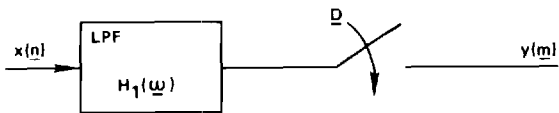


Fig. 4. A decimator for decimating by the matrix factor  $D$ .

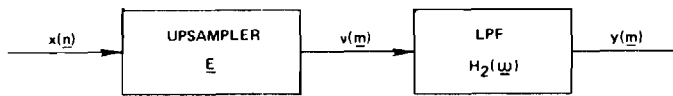


Fig. 5. An interpolator for interpolating by the matrix factor  $E$ .

lattice. This represents a degenerate case in which decimation becomes simply a reordering of the input samples. In the more interesting case  $|\det D| \geq 2$ . Let  $I_{VD}$  denote the region containing one period of  $Y(\omega)$ . It is clear that  $I_{VD} \subset I_V$ . In fact, the volume of  $I_{VD}$  will be smaller than the volume of  $I_V$  by a factor of  $|\det D|$ . If  $x_a(t)$  is recoverable from  $y(m)$  then we must also have  $I_X \subset I_{VD}$ . This leads to the structure for a decimator which is shown in Fig. 4. The signal  $x(n)$  is passed through a low-pass filter whose passband is confined to  $I_{VD}$  and is then downsampled by the decimation matrix  $D$ . The frequency response of the low-pass filter is given by

$$H(\omega) = \begin{cases} \frac{1}{|\det D|}, & \omega \in I_{VD} \\ 0, & \text{otherwise.} \end{cases} \quad (49)$$

Interpolation is the reverse problem by which we try to increase the number of samples taken from an underlying band-limited analog signal by filtering. Let  $E$  be a matrix of integers, known as the *interpolation matrix*, such that

$$x(n) = y(En). \quad (50)$$

Here  $x(n)$  is the original low-rate signal and  $y(n)$  is the higher rate interpolated signal. Let  $V$  be the sampling matrix for  $y(m)$ ; then  $EV$  is the sampling matrix for  $x(n)$ . From our earlier results it then follows that  $X(\omega)$  is periodic with periodicity matrix  $2\pi(EV)^T^{-1}$  and that  $Y(\omega)$  is periodic with periodicity matrix  $2\pi(V^T)^{-1}$ . If  $I_{EV}$  and  $I_V$  denote one period of the respective Fourier transforms, then

$$I_X \subset I_{EV} \subset I_V.$$

The regions of support for  $X(\omega)$  and  $Y(\omega)$  are both  $I_X$  which is a subset of their respective periods.

Interpolation can be performed using the two-step procedure which is illustrated in Fig. 5. The sequence  $x(n)$  is first passed through an up-sampler whose operation is described by the rule

$$v(m) = \begin{cases} x(n), & m = En \\ 0, & \text{otherwise.} \end{cases} \quad (51)$$

The signal  $v(m)$  has the proper lattice geometry, but since

$$V(\omega) = \sum_m v(m) e^{-j\omega^T V m} \quad (52)$$

$$\begin{aligned} &= \sum_n v(En) e^{-j\omega^T V En} \\ &= \sum_n x(n) e^{-j\omega^T V En} = X(\omega) \end{aligned} \quad (53)$$

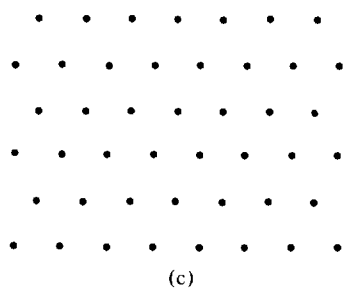
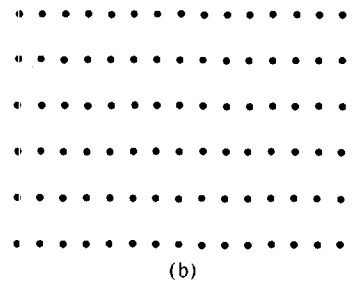
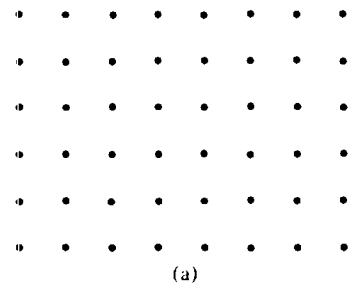


Fig. 6. An example of resampling. (a) Original sampling lattice. (b) High density intermediate lattice. (c) Final sampling lattice.



Fig. 7. A system for performing a general change of sampling lattice using an interpolator and a decimator.

it is periodic with periodicity matrix  $2\pi(VE)^T^{-1}$ . This signal must then be passed through a low-pass filter with the frequency response

$$H(\omega) = \begin{cases} |\det E|, & \omega \in I_{EV} \\ 0, & \text{otherwise} \end{cases} \quad (54)$$

to remove all periods except for the central one.

As an example, let us consider the interpolation from the rectangular lattice shown in Fig. 6(a) to the hexagonal lattice shown in Fig. 6(c). This can be accomplished by first interpolating the rectangularly sampled signal to the lattice shown in Fig. 6(b) and then decimating the result. The overall system is shown in Fig. 7. There we have cascaded an interpolator, and a decimator; the two low-pass filters have been combined into a single low-pass filter. One appropriate sampling matrix for the lattice in Fig. 6(a) is

$$V_R = \begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix}. \quad (55)$$

Similarly, for the intermediate lattice and the hexagonal lattice we can use

$$V_I = \begin{bmatrix} \frac{T_1}{2} & 0 \\ 0 & T_2 \end{bmatrix} \quad (56)$$

$$V_{II} = \begin{bmatrix} T_1 & \frac{T_1}{2} \\ 0 & T_2 \end{bmatrix}. \quad (57)$$

A sufficient pair of interpolation and decimation matrices to accomplish the desired transformations are given by

$$E = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \quad (58)$$

$$D = \begin{bmatrix} 2 & -1 \\ 0 & 1 \end{bmatrix}. \quad (59)$$

#### SUMMARY

In this paper we have presented a very general framework for processing multidimensional signals which have been sampled on any given periodic sampling lattice. The general conclusion seems to be that almost anything that can be done in the one-dimensional case or the rectangular multidimensional case can be generalized. This notation has been greatly aided by the introduction of a number of simple matrix operators which act on the indexes of the signal. Not only do these matrices help to make the analogy with the 1-D case clearer, but they provide a compact means of describing the geometry of a sampling lattice. It is of particular interest to note that the problem of interpolating from one lattice to another can be handled simply with generalized decimators and interpolators.

#### REFERENCES

- [1] R. M. Mersereau, "The processing of hexagonally sampled two-dimensional signals," *Proc. IEEE*, vol. 67, pp. 930-949, May 1979.
- [2] R. M. Mersereau and T. C. Speake, "A unified treatment of Cooley-Tukey algorithms for the evaluation of the multi-dimensional DFT," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 1011-1018, Oct. 1981.
- [3] D. P. Petersen and D. Middleton, "Sampling and reconstruction of wavenumber-limited functions in  $N$ -dimensional Euclidean spaces," *Inform. Contr.*, vol. 5, pp. 279-323, 1962.

- [4] T. C. Speake and R. M. Mersereau, "An interpolation technique for periodically sampled two-dimensional signals," in *1981 IEEE Int. Conf. Acoust., Speech, Signal Processing*, Apr. 1981, pp. 1010-1013.
- [5] C. G. Lekkerkerker, *Geometry of Numbers*. Groningen: The Netherlands: Wolters-Noordhoff, 1969.
- [6] R. W. Schafer and L. R. Rabiner, "A digital signal processing approach to interpolation," *Proc. IEEE*, vol. 61, pp. 692-702, June 1973.



**Russell M. Mersereau** (S'69-M'73-SM'78-F'82) was born in Cambridge, MA, on August 29, 1946. He received the S.B., S.M., and Sc.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1969, 1969, and 1973, respectively.

From 1971 to 1973 he was an instructor in the Department of Electrical Engineering, M.I.T., and from 1973 to 1975 he was with the Research Laboratory of Electronics and Department of Electrical Engineering as a Research

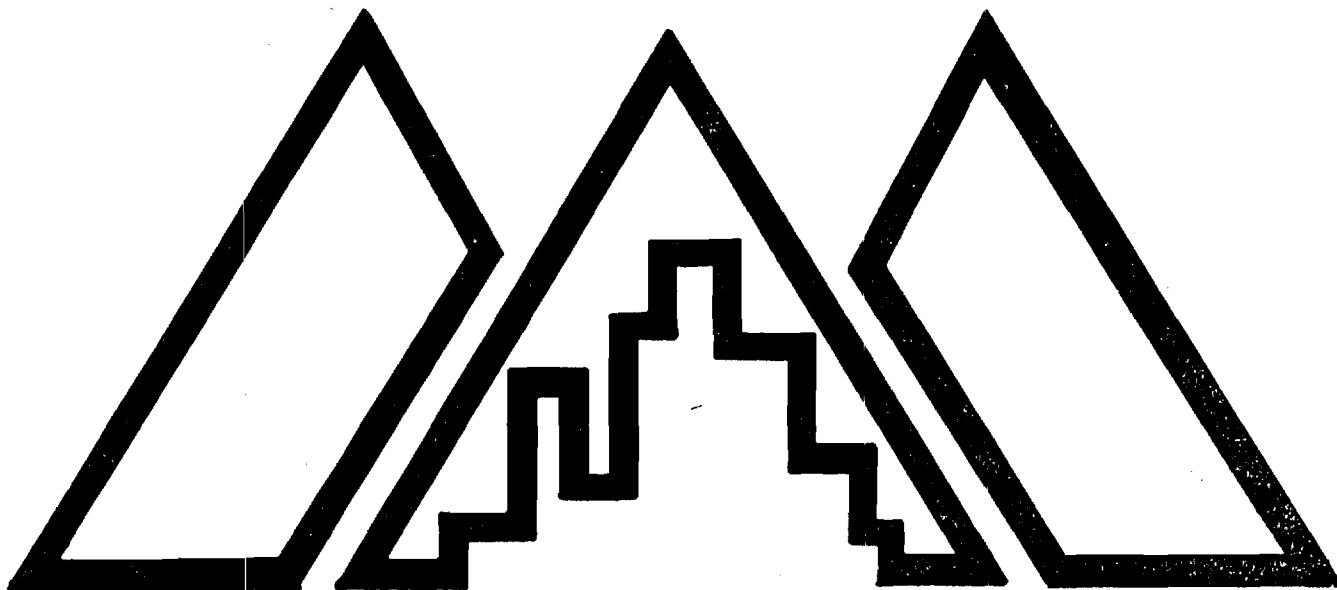
Associate. Currently, he is an Associate Professor in the School of Electrical Engineering, Georgia Institute of Technology, Atlanta, where his primary interests are digital processing of multidimensional signals.

Dr. Mersereau is a member of Tau Beta Pi, Eta Kappa Nu, and Sigma Xi. He was the corecipient (with D. E. Dudgeon) of the 1976 Browder J. Thompson Memorial Prize from the IEEE and the recipient of the 1977 Research Unit Award from the American Society for Engineering Education Southeastern Section. He received a teaching award from the School of Electrical Engineering in 1978. He was formerly the Associate Editor for Signal Processing of the IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING and Technical Chairman of ICASSP '81, and he is currently a member of the ASSP AdCom and the Technical Committees on Digital Signal Processing and Multidimensional Digital Signal Processing.



**Theresa C. Speake** was born in Atlanta, GA, on September 18, 1952. She received the B.E.E. degree in electrical engineering from the Georgia Institute of Technology, Atlanta, in 1974, and the M.S. degree in electrical engineering from Stanford University, Stanford, CA, in 1976.

From 1975 to 1976, she was a Senior Engineer at Sylvania Electronics Systems Group, Mountain View, CA. In 1976 she returned to Georgia Tech, where she was awarded a President's Fellowship to continue her graduate education. From 1977 through 1981 she was employed at Georgia Tech as a Research and Teaching Assistant. Since January 1982, she has been an Assistant Professor in the Electrical Engineering Technology Department, Southern Technical Institute, Marietta, GA.



**ICASSP 80**

**PROCEEDINGS**

**Volume 3 of 3**

**IEEE INTERNATIONAL CONFERENCE  
ON  
ACOUSTICS, SPEECH AND  
SIGNAL PROCESSING**

80CH1559-4

A COMPARISON OF HEXAGONALLY AND RECTANGULARLY-SAMPLED  
TWO-DIMENSIONAL FIR DIGITAL FILTERS\*

Russell M. Mersereau, Tae H. Joo and Theresa C. Speake

School of Electrical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332

ABSTRACT

This paper presents a comparison of two-dimensional finite area impulse response (FIR) digital filters designed using three popular design methodologies - through the use of windows, through the use of transformations of one-dimensional designs, and through the use of optimal Chebyshev design techniques. In addition the comparison includes filters designed for processing both rectangularly-sampled and hexagonally-sampled data. The filters are compared with respect to ease of design and the efficiency of the resulting implementation.

INTRODUCTION

In (1), Mersereau showed that familiar signal processing algorithms for rectangularly-sampled signals could be modified for the processing of hexagonally-sampled signals, and he claimed that implementing the hexagonal algorithms should require less computational effort than implementing comparable rectangular algorithms. The purpose of this paper is to test this hypothesis for FIR digital filtering. To this end, three common design methodologies for 2-D FIR filter design are extended to the hexagonal case. The resulting designs are used to design filters to a common set of specifications and the resulting filters are compared.

DESIGN METHODS

The impulse response  $h(m,n)$  and the frequency response  $H(\omega_1, \omega_2)$  of a linear, shift-invariant filter are Fourier transforms of each other. Thus in the rectangular case we have

$$H(\omega_1, \omega_2) = \sum_m \sum_n h(m,n) \exp[-j(m\omega_1 + n\omega_2)] \quad (1)$$

and in the hexagonal case

$$H(\omega_1, \omega_2) = \sum_m \sum_n h(m,n) \exp\left[\frac{2m-n}{\sqrt{3}} \omega_1 + n\omega_2\right]. \quad (2)$$

\* This work was supported in part by the National Science Foundation under grant ENG78-17201 and in part by the Joint Services Electronics Program under contract No. DAAG29-78-C-0005.

The Window Method

The window method for FIR filter design is well-understood in both the rectangular (1) and hexagonal cases (2). If  $i(m,n)$  is the ideal rectangular impulse response, then  $h(m,n)$  can be chosen in one of two ways. Either

$$h(m,n) = i(m,n) w(m) w(n) \quad (3)$$

or

$$h(m,n) = i(m,n) w(\sqrt{m^2 + n^2}). \quad (4)$$

Eq. (3) is known as the outer product formulation and (4) represents the symmetric or Huang formulation. In both cases  $w(\cdot)$  is any one-dimensional window function, but in our experiments a Kaiser window was used.

For hexagonal filters the outer product formulation is

$$h(m,n) = i(m,n) w(m) w(n) w(m-n) \quad (5)$$

and the Huang formulation becomes

$$h(m,n) = i(m,n) w(2\sqrt{(m^2 + n^2 - mn)/3}) \quad (6)$$

where  $i(m,n)$  now represents the ideal hexagonal impulse response.

The Transformation Method

With the transformation method a one-dimensional zero-phase FIR filter, whose frequency response can be written in the form

$$H(\omega) = \sum_n a(n) \cos n\omega = \sum_n a(n) T_n(\cos \omega) \quad (7)$$

is converted into a two-dimensional filter by means of a substitution of variables (3,4)

$$F(\omega_1, \omega_2) \rightarrow \cos \omega. \quad (8)$$

For a first-order transformation  $F$  is of the form

$$F(\omega_1, \omega_2) = A + B\cos\omega_1 + C\cos\omega_2 + D\cos(\omega_1 + \omega_2) + E\cos(\omega_1 - \omega_2) \quad (9)$$

in the rectangular case and

$$F(\omega_1, \omega_2) = A + B\cos\frac{2\omega_1}{\sqrt{3}} + C\cos\left(\frac{\omega_1}{\sqrt{3}} + \omega_2\right) + D\cos\left(\frac{\omega_1}{\sqrt{3}} - \omega_2\right) \quad (10)$$

in the hexagonal one. The choice of A, B, C, D, and E affects the shape of the passbands and stopbands of the 2-D filter, while the 1-D filter coefficients  $a(n)$  affect the filter order and the size of the passband and stopband ripples.

### Equiripple Designs

Equiripple filters are designed to minimize the Chebychev error norm

$$E = \max_{(\omega_1, \omega_2) \in K} |[I(\omega_1, \omega_2) - H(\omega_1, \omega_2)]W(\omega_1, \omega_2)| \quad (11)$$

for a filter with a given region of support, by suitably choosing the impulse response coefficients. In this study this was accomplished using the methods of (5), with the obvious modifications made for the hexagonal case.

### THE FILTER PROTOTYPES

Rectangular and hexagonal FIR filters were designed to satisfy one of four sets of filter specifications. The first three prototypes corresponded to circularly symmetric lowpass filters of the general form.

$$|H(\omega_1, \omega_2) - 1| \leq \delta_p, \quad \omega_1^2 + \omega_2^2 \leq \omega_p^2 \quad (12)$$

$$|H(\omega_1, \omega_2)| \leq \delta_s, \quad \omega_1^2 + \omega_2^2 \geq \omega_s^2 \quad (13)$$

These ideal specifications are illustrated in Figures 1 and 2 for rectangular and hexagonal filters, respectively.

- Prototype 1:  $\omega_p = .36\pi, \omega_s = .5\pi,$   
 $\delta_p = .05, \delta_s = .05$   
 Prototype 2:  $\omega_p = .3\pi, \omega_s = .5\pi,$   
 $\delta_p = .05, \delta_s = .01$   
 Prototype 3:  $\omega_p = .3, \omega_s = .5\pi,$   
 $\omega_p = .02, \omega_s = .02$

The fourth prototype filter was a four-quadrant symmetric fan filter with a fan angle of  $45^\circ$  in the rectangular case and  $60^\circ$  in the hexagonal case, as shown in Figures 3 and 4.

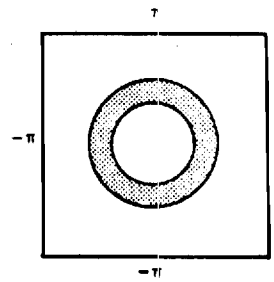


Figure 1

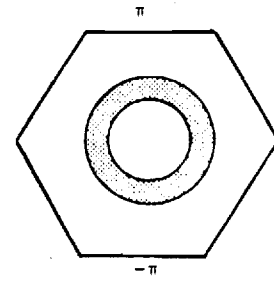


Figure 2

The passband and stopband regions for rectangularly- and hexagonally-sampled prototypes 1, 2, and 3.

Prototype 4:  $\omega_p, \omega_s$  (see Figs. 3 and 4);  
 $\delta_p = \delta_s = .02$

While designing the two fans to have different fan angles means that we are not comparing equivalent filters, in both bases we are looking at fans that are equally well-conditioned to their sampling strategy.

Filters were designed which would meet these specifications exactly or exceed them, subject to the constraint that the filter diameter be an odd integer. In all cases the passband and stopband cutoffs were satisfied exactly and any excess filtering capability was used to reduce  $\delta_p$  and  $\delta_s$ .

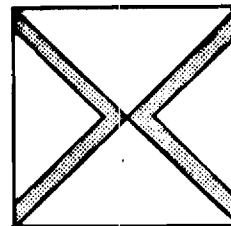


Figure 3

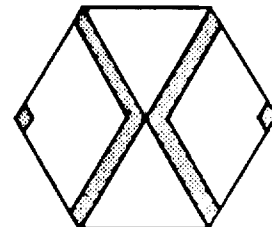


Figure 4

The passband and stopband regions for rectangularly- and hexagonally-sampled prototype 4.

### COMPUTATIONAL EFFICIENCY

The convolution sum provides one implementation for a 2-D FIR filter. With this implementation, called the direct implementation, the number of additions required is equal to the number of nonzero samples in the impulse response  $h(m, n)$ . The number of multiplications required is equal to the number of distinct coefficients in  $h$  which is always less than the number of

coefficients if the filter is zero-phase and the number of distinct coefficients is further reduced if the filter impulse response is symmetric. The amount of storage required is enough to hold a number of rows of  $x$  equal to the number of non-zero rows of  $h$ . Only one I/O pass through the data is required. For the purposes of this comparison, complexity for the window and equiripple designs is measured by the number of multiplications required for each sample of the output array. For hexagonal designs this number is multiplied by .866 to account for the fact that the hexagonal output array has 13.4% fewer samples to compute.

Filters designed by the transformation method possess a special cascade-like implementation which is described in (4) for the rectangular-case and (1) for the hexagonal one. The number of multiplications depends upon the exact transformation functions  $F(\omega_1, \omega_2)$  used, but it is less than the number required in a direct implementation. For a filter with radius  $R$  the number of multiplies/output sample falls in the range

$$R + 1 \leq \text{comp} \leq 6R + 1 \quad (14)$$

in the rectangular case and

$$.866(R + 1) \leq \text{comp} \leq .866(5R + 1) \quad (15)$$

in the hexagonal case where once again the correction factor of .866 is used.

FIR filters can also be implemented by Fourier transform methods. Such implementations are not included in this comparison for a number of reasons. Among these are the fact that the complexity depends upon the size of the output array, it is essentially independent of filter order, and it is not easily measured since storage and I/O handling become important considerations. Nevertheless these methods represent the implementation of choice for high-order filters.

We have also not directly compared design efficiencies although it should be noted that window and transformation designs are extremely simple to perform and equiripple designs are extremely difficult. In fact we were not able to design equiripple approximation to Prototype 4. The number of degrees of freedom required for the approximation was too high.

## RESULTS

The results of the comparison are shown in the tables. Table I shows the comparisons of the twelve circular lowpasses designed using windows (3 prototypes x 2 sampling strategies x 2 window formulations). For each the filter radius is given along with the passband and stopband ripples, and the number of multiplications per output point (corrected for the hexagonal filters). It should be noted that for a given filter radius the hexagonal designs involve fewer multiplies than the rectangular ones and the Huang formulations require fewer multiplications than the outer product formulations. This is because the Huang filters have

a circular region of support. More specifically, for these prototypes the hexagonal filters afford a savings in multiplications of about 56% over their rectangular equivalents. Huang formulations provide a savings in multiplications of about 16% over outer product formulations.

Filter	Radius	$\delta_p$	$\delta_s$	implementation complexity
Rect 1 Huang	10	.039	.042	56
Rect 1 OP	10	.044	.047	66
Rect 2 Huang	12	.014	.001	76
Rect 2 OP	12	.009	.008	91
Rect 3 Huang	10	.019	.016	56
Rect 3 OP	10	.020	.015	66
Hex 1 Huang	11	.044	.042	29
Hex 1 OP	10	.050	.038	31
Hex 2 Huang	11	.016	.010	29
Hex 2 OP	11	.014	.001	36
Hex 3 Huang	10	.019	.014	24
Hex 3 OP	10	.020	.014	31

TABLE I.

Results for window designs for lowpass filters

The results for the six equiripple lowpass designs are summarized in Table II. Here it is interesting to note that the rectangular and hexagonal approximations to each prototype are virtually identical in terms of required radius and passband and stopband ripple, but that the hexagonal designs can be implemented with about one-half the multiplies. The exact savings here is 52%. The additional column in the table gives the number of degrees of freedom required for the design. This is a crude measure of design complexity. The total design time is roughly proportional to the third or fourth power of the number of degrees of freedom. Thus the rectangular filters required approximately an order of magnitude longer to design than the hexagonal filters. For design times measured in hours this savings is significant.

The transformation designs for the circular lowpass prototypes are summarized in Table III. The transformation parameters used in the rectangular case were  $A = -.5$ ,  $B = C = .5$ ,  $D = E = .25$  and in the hexagonal case  $A = -1/3$ ,  $B = C = D = 4/9$ . It will be noted that the filters are similar and

Filter	Radius	$\delta_p$	$\delta_s$	DOF	implementation complexity
Rect 1	8	.047	.047	45	45
Rect 2	9	.035	.007	55	55
Rect 3	9	.012	.012	55	55
Hex 1	8	.048	.048	25	22
Hex 2	9	.035	.007	30	26
Hex 3	9	.012	.012	30	26

TABLE II

Results for Equiripple Designs for lowpass filters  
DOF = Degrees of Freedom

Filter	Radius	$\delta_p$	$\delta_s$	implementation complexity
Rect. 1	11	.040	.040	35
Rect. 2	9	.045	.009	29
Rect. 3	9	.015	.015	29
Hex 1	9	.046	.046	24
Hex 2	9	.050	.010	24
Hex 3	9	.020	.020	24

TABLE III

Results for transformation designs for lowpass filters

and that the hexagonal designs offer a slight reduction in computation (avg. = 22%). The savings here are less than with the other designs. This is due to the specific rectangular transformation chosen, which is a particularly simple one to implement.

Table IV summarizes the two fan filters. One fact which is immediately apparent is that these require more multiplications for their implementation than the lowpasses. This is because the filter responses are less symmetric and thus there is less redundancy among the coefficients to be exploited. We also see that here the rectangular transformation design is easier to implement than the hexagonal one even though the latter is of lower order. Once again this was due to the particularly benign specific transformation used ( $A = D = E = 0$ ,  $B = -C = .5$ ). The hexagonal transformation is ( $A = -.28$ ,  $B = .64$ ,  $C = D = -.32$ ).

It should also be noted that for this prototype the outer product windows are to be preferred over the Huang windows. The circular region of support afforded by the Huang windows is not the natural one for a fan filter. In both the rectangular and hexagonal cases the window designs require significantly lower order filters than the transformation method. Thus the window designs will require only about 1/3 the storage of the transformation designs. In the hexagonal case the required computation is also less.

Type	Order (Radius)	$\delta_p$	$\delta_s$	implementation complexity
Rect Huang	11	.011	.019	121
Rect OP	8	.018	.016	81
Rect trans	27	.019	.019	55
Hex Huang	11	.013	.019	74
Hex OP	9	.018	.016	65
Hex trans	23	.018	.018	81

TABLE IV

Results of the fan filter comparison

Looking across the tables we see for the circular lowpass prototypes in the hexagonal case there is little to choose between the three design methods with respect to the implementation complexity of the resulting filters, although the design complexity of the equiripples makes them somewhat unattractive. In the rectangular case there is a savings in using the transformation method. Similar conclusions can be drawn for the two fan filters. There would also appear to be clear computational savings in using hexagonal filters over rectangular ones.

The results of this comparison seem to suggest three things. (1) The transformation method is probably the design method of choice for rectangular lowpass filters. (2) Hexagonal filters would appear to offer substantial savings over rectangular filters for many sets of filter specifications. (3) Conclusions drawn for circularly symmetric lowpasses may not be too useful for other types of filters.

#### REFERENCES

- (1) R.M. Mersereau, "The processing of hexagonally-sampled two-dimensional signals," *Proc. IEEE*, 67, pp 930-949, June 1979.
- (2) T.C. Speake and R.M. Mersereau, "A comparison of different window formulations for two-dimensional FIR filter design," 1979 IEEE International Conference on Acoustics, Speech and Signal Processing Record, pp 5-8.
- (3) R.M. Mersereau, W.F.G. Mechlbrauker, and T.F. Quatieri, Jr., "McClellan transformations for two-dimensional digital filtering: I. Design," *IEEE Trans. Circuits Syst.*, v. CAS-23, pp 405-414, July 1976.
- (4) J.H. McClellan and D.S.K. Chan, "A 2-D FIR filter structure derived from the Chebyshev recursion," *IEEE Trans. Circuits Syst.*, v. CAS-24, pp 372-378, July 1977.

# Lecture Notes in Control and Information Sciences

Edited by A.V. Balakrishnan and M. Thoma

**IRIA**

28

## Analysis and Optimization of Systems

Proceedings of the Fourth International  
Conference on Analysis and Optimization  
of Systems

Versailles, December 16-19, 1980

Edited by  
A. Bensoussan and J.L. Lions



Springer-Verlag

GENERALIZED COOLEY-TUKEY ALGORITHMS FOR EVALUATION OF  
MULTI-DIMENSIONAL DISCRETE FOURIER TRANSFORMS\*

Russell M. Mersereau  
Theresa C. Speake

School of Electrical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332

ABSTRACT

In this paper the Cooley-Tukey fast Fourier transform (FFT) algorithm is generalized to the multi-dimensional case in a natural way that incorporates the standard row-column and vector-radix algorithms as special cases. It can be used for the evaluation of discrete Fourier transforms of rectangularly or hexagonally sampled signals or signals which are arbitrarily sampled in either the spatial or frequency domain. These fast Fourier transform algorithms are shown to result from the factorization of an integer matrix; different algorithms correspond to different factorizations. This paper will first derive a generalized discrete Fourier transform, then derive the general Cooley-Tukey algorithm, and conclude by interpreting existing multi-dimensional FFT algorithms in terms of the generalized one.

This work was supported, in part, by the National Science Foundation under grant ECS-7817201 and by the Joint Services Electronics Program under Contract DAAG29-76-G-0226.

I. INTRODUCTION

Numerous applications of digital signal processing call for the evaluation of discrete Fourier transforms of multi-dimensional sequences. For example, this evaluation is fundamental to the implementation of multi-dimensional FIR filters for multi-dimensional spectral analysis, and to the transform coding of images. Usually these transforms are computed by means of one of two procedures -- through the use of a row-column decomposition of the discrete Fourier transform summation into a number of one-dimensional DFT's, or through the use of a vector-radix decomposition of the DFT sum [1]. The latter procedure was discovered by Rivard [2], but it has been independently derived by a number of other researchers [3,4,5].

In this presentation it is shown that these two algorithms and a host of other multi-dimensional fast Fourier transforms (FFT's) are special cases of a generalization of the one-dimensional Cooley-Tukey FFT algorithm [6] to the multi-dimensional case. This generalized algorithm allows the efficient evaluation of a large number of discrete Fourier transforms including DFT's of rectangularly sampled sequences, hexagonally sampled sequences [7], and arbitrary periodically sampled sequences. It can even be used to evaluate DFT's which relate rectangularly sampled sequences to their hexagonally sampled Fourier transforms.

The key to the efficiency of the 1-D FFT is the fact that significant computational savings can be realized when the length of the transform,  $N$ , is a highly composite number. The key to the efficiency of the multi-dimensional FFT is the fact that significant computational savings can be realized when the periodicity matrix,  $\underline{N}$ , which is an integer matrix that depends upon the support of the sequence, is a highly composite matrix. Alternate factorizations of this matrix lead to different FFT algorithms -- including the row-column decomposition and the vector-radix algorithm.

This paper is divided into three parts. First, the periodicity matrix is defined and the discrete Fourier transform is derived. Then the multi-dimensional Cooley-Tukey algorithm is presented and finally some specific factorizations of the periodicity matrix are considered.

II. A GENERAL DISCRETE FOURIER TRANSFORM

2.1 Periodic Sequences

A periodic two-dimensional sequence is one which repeats itself at regularly spaced intervals. However, in the 2-D case such a sequence must repeat in two different directions at once, which makes the formal definition of two-dimensional periodicity somewhat more complex than that of one-dimensional periodicity.

A 2-D sequence,  $\bar{x}(n_1, n_2)$ , is periodic if it satisfies the following conditions for all integer pairs  $(n_1, n_2)$ :

$$\bar{x}(n_1 + N_{11}, n_2 + N_{21}) = \bar{x}(n_1, n_2) \quad (1)$$

$$\bar{x}(n_1 + N_{12}, n_2 + N_{22}) = \bar{x}(n_1, n_2) \quad (2)$$

$$N_{11}N_{22} - N_{12}N_{21} \neq 0.$$

Using matrix notation, this condition can be abbreviated as

$$\bar{x}(\underline{n} + \underline{N}\underline{r}) = \bar{x}(\underline{n}) \quad (3)$$

$$\det(\underline{N}) \neq 0.$$

Here  $\underline{n}$  and  $\underline{r}$  are column vectors with integer entries and  $\underline{N}$ , which is called the periodicity matrix, is a 2 x 2 integer matrix. For a given periodic sequence the periodicity matrix is not unique.

The condition in (3) works equally well to define an M-dimensional periodic sequence; in this case the periodicity matrix is an M x M matrix of integers. The columns of  $\underline{N}$  are vectors which represent the displacements from any sample of the sequence  $\bar{x}$  to its equivalent samples on M other periods. The requirement that  $\det(\underline{N})$  be non-zero imposes the requirement that the sequence repeat in M independent directions. This determinant has another interesting property, however;  $|\underline{N}| = |\det(\underline{N})|$  is the number of samples in one period of the sequence  $\bar{x}$ . If  $\underline{N}$  is diagonal, we will say that  $\bar{x}$  is rectangularly periodic. This is the most frequently studied case.

## 2.2 Discrete Fourier Series Representations for Multi-Dimensional Periodic Sequences

Consider a periodic sequence  $\bar{x}(\underline{n})$  which has the periodicity matrix  $\underline{N}$  and let  $I_{\underline{N}}$  denote a region in the n-domain which contains exactly one period of this sequence. This region will be denoted as the fundamental period of the array; it contains exactly  $|\underline{N}|$  samples of  $\bar{x}$ .

By analogy with the one-dimensional case, it is not unreasonable to hypothesize that  $\bar{x}(\underline{n})$  can be uniquely represented as a finite sum of harmonically related complex exponentials. Such a representation assumes the form

$$\bar{x}(\underline{n}) = \sum_{\underline{k} \in J_{\underline{N}}} a(\underline{k}) \exp[j\underline{k}'\underline{R}\underline{n}] \quad (4)$$

where  $\underline{k}$  and  $\underline{n}$  are integer vectors,  $J_{\underline{N}}$  is a region of finite extent in the  $\underline{k}$ -domain, and the apostrophe denotes the operation of matrix (or vector) transposition. Since  $\tilde{x}$  is periodic it follows that

$$\begin{aligned}\tilde{x}(\underline{n}) &= \tilde{x}(\underline{n} + \underline{N}\underline{r}) = \sum_{\underline{k} \in J_{\underline{N}}} a(\underline{k}) \exp[\underline{j}\underline{k}'\underline{R}(\underline{n} + \underline{N}\underline{r})] \\ &= \sum_{\underline{k} \in J_{\underline{N}}} a(\underline{k}) \exp[\underline{j}\underline{k}'\underline{R}\underline{N}\underline{r}] \exp[\underline{j}\underline{k}'\underline{R}\underline{n}].\end{aligned}\quad (5)$$

The right-hand sides of (4) and (5) must be equal for all values of  $\underline{n}$ . In particular they must be equal for the  $|\underline{N}|$  independent samples of  $\tilde{x}(\underline{n})$  contained in  $I_{\underline{N}}$ . If at least  $|\underline{N}|$  of the complex exponentials  $\exp[\underline{j}\underline{k}'\underline{R}\underline{n}]$  are linearly independent (a condition which will be verified below), then it follows that such an equality demands

$$\exp[\underline{j}\underline{k}'\underline{R}\underline{N}\underline{r}] = 1 \quad (6)$$

for all integer vectors  $\underline{r}$  and  $\underline{k}$ . This condition, in turn, will be true whenever

$$\underline{R} = 2\pi\underline{N}^{-1}. \quad (7)$$

With the notational change

$$a(\underline{k}) = \frac{1}{|\underline{N}|} \tilde{X}(\underline{k})$$

equation (4) becomes

$$\tilde{x}(\underline{n}) = \frac{1}{|\underline{N}|} \sum_{\underline{k} \in J_{\underline{N}}} \tilde{X}(\underline{k}) \exp[\underline{j}\underline{k}'(2\pi\underline{N}^{-1})\underline{n}]. \quad (8)$$

The complex exponentials which appear in this sum are periodic in both  $\underline{n}$  (with periodicity matrix  $\underline{N}$ ) and  $\underline{k}$  (with periodicity matrix  $\underline{N}'$ ). Therefore, at most  $|\underline{N}'| = |\underline{N}|$  of these exponentials can be linearly independent.

The coefficients  $\tilde{X}(\underline{k})$  can be evaluated from the sequence  $\tilde{x}(\underline{n})$  using the expression

$$\tilde{X}(\underline{k}) = \sum_{\underline{n} \in I_{\underline{N}}} \tilde{x}(\underline{n}) \exp[-\underline{j}\underline{k}'(2\pi\underline{N}^{-1})\underline{n}]. \quad (9)$$

With this definition, the coefficients  $\tilde{X}(\underline{k})$  can be seen to form a periodic sequence

with the periodicity matrix  $\underline{N}$ '. If the region  $J_{\underline{N}}$  is chosen to contain exactly one period of this sequence, then (8) and (9) are seen to constitute a mathematical identity. If the  $\{\underline{k}\}$  are chosen from  $J_{\underline{N}}$  and the  $\{\underline{n}\}$  are chosen from  $I_{\underline{N}}$ , then the complex exponentials  $\exp[j\underline{k}'2\pi\underline{N}^{-1}\underline{n}]$  are orthogonal and linearly independent over both  $I_{\underline{N}}$  and  $J_{\underline{N}}$ . Equations (8) and (9) constitute a discrete Fourier series representation for a multi-dimensional periodic sequence.

### 2.3 A General Multi-Dimensional Discrete Fourier Transform

Let  $x(\underline{n})$  be a sequence with finite support confined to the region  $I_{\underline{N}}$ . This sequence can be periodically extended to form the periodic sequence  $\tilde{x}(\underline{n})$  by means of the relation

$$\tilde{x}(\underline{n}) = \sum_{\underline{r}} x(\underline{n} + \underline{N}\underline{r}) . \quad (10)$$

Because the sequence  $x(\underline{n})$  has support limited to  $I_{\underline{N}}$ , it can be recovered from  $\tilde{x}$ . Specifically, it follows that

$$x(\underline{n}) = \begin{cases} \tilde{x}(\underline{n}) & , \underline{n} \in I_{\underline{N}} \\ 0 & , \text{otherwise} \end{cases} . \quad (11)$$

There is thus a one-to-one correspondence between sequences with support on  $I_{\underline{N}}$  and periodic sequences with periodicity matrix  $\underline{N}$ . Because of this one-to-one correspondence, it follows that  $x(\underline{n})$  can be exactly represented by the Fourier series coefficients  $\tilde{X}(\underline{k})$ . This relationship is the discrete Fourier transform (DFT) representation of a sequence with support on  $I_{\underline{N}}$ .

$$X(\underline{k}) = \sum_{\underline{n} \in I_{\underline{N}}} x(\underline{n}) \exp[-j\underline{k}'(2\pi\underline{N}^{-1})\underline{n}] , \quad \underline{k} \in J_{\underline{N}} \quad (12)$$

$$x(\underline{n}) = \frac{1}{|\underline{N}|} \sum_{\underline{k} \in J_{\underline{N}}} X(\underline{k}) \exp[j\underline{k}'(2\pi\underline{N}^{-1})\underline{n}] , \quad \underline{n} \in I_{\underline{N}} \quad (13)$$

In addition to their interpretation as Fourier series coefficients, the numbers  $X(\underline{k})$  can be interpreted in terms of samples of the Fourier transform of the sequence  $x(\underline{n})$ , which, in turn, can be interpreted in terms of the Fourier transform of a continuous, bandlimited signal which has been periodically sampled. Let  $x_c(t)$  denote a continuous, bandlimited signal whose Fourier transform is confined to the region  $I_{\underline{U}}$  and let  $x(\underline{n})$  be the sequence derived from that signal by periodic

pling with a sampling matrix  $\underline{V}$ . Thus

$$\underline{x}(n) = \underline{x}_a(\underline{V}n) . \quad (14)$$

If  $\underline{V}$  is such that no aliasing has occurred, then the Fourier transform of the continuous signal,  $X_a(\underline{\omega})$  is expressible as

$$X_a(\underline{\omega}) = \begin{cases} |\underline{V}| \sum_n \underline{x}(n) \exp[-j\underline{\omega}'\underline{V}n] , & \underline{\omega} \in I_U \\ 0 , & \text{otherwise} \end{cases} \quad (15)$$

where  $\underline{\omega}$  is a column vector of continuous frequency variables. Comparing eqs. (12) and (15) it is seen that

$$X(k) = \frac{1}{|\underline{V}|} X_a((\underline{V}^{-1})'(2\pi\underline{N}^{-1})'k) . \quad (16)$$

The locations of these samples in the Fourier domain depend upon both the sampling matrix  $\underline{V}$  and the periodicity matrix  $\underline{N}$ , which must be an integer matrix.

Some special cases of (12) are of sufficient importance to justify elucidation. Rectangular sampling, for example, corresponds to the special case where  $\underline{V}$  is diagonal. If the Fourier transform is also to be rectangularly sampled, a diagonal periodicity matrix is required, which in the two-dimensional case assumes the form

$$\underline{N} = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} . \quad (17)$$

For this case the regions  $I_N$  and  $J_N$  can be chosen to be  $N_1 \times N_2$  point rectangular regions, although this choice is not necessary. The periodicity matrix

$$\underline{N} = \begin{bmatrix} N_1 & N_1 \\ N_2 & -N_2 \end{bmatrix} \quad (18)$$

will give hexagonal samples of the Fourier transform of a rectangularly sampled sequence. Here  $I_N$  and  $J_N$  can be a  $2N_1 \times N_2$  point rectangular region. The periodicity matrix

$$\underline{N} = \begin{bmatrix} N_1 & -1 \\ 0 & N_2 \end{bmatrix} \quad (19)$$

converts the 2-D DFT into what is essentially a one-dimensional DFT [8]. Here the samples of the Fourier transform lie at the points

$$\omega_1 = \frac{2\pi k}{N_1} ; \omega_2 = \frac{2\pi k}{N_1 N_2} , \quad k = 0, 1, \dots, N_1 N_2 - 1 . \quad (20)$$

Hexagonal sampling of an analog signal and its Fourier transform corresponds to a sampling matrix of the form

$$\underline{V} = \begin{bmatrix} T_1 & T_1 \\ T_2 & -T_2 \end{bmatrix} \quad (21)$$

and a periodicity matrix of the form

$$\underline{N} = \begin{bmatrix} N_1 & N_2 \\ N_2 & N_1 \end{bmatrix} \quad (22)$$

All of the essentially different DFT's have the form of eq (12); all that is different is the specification of the periodicity matrix  $\underline{N}$ . In the next section an algorithm will be presented for evaluating (12). That algorithm can be used for all of the specific DFT's presented above.

### III. EFFICIENT COMPUTATION OF THE DFT

In this section we shall consider efficient algorithms for the evaluation of a DFT of the form

$$X(\underline{k}) = \sum_{\underline{n} \in I_{\underline{N}}} x(\underline{n}) \exp[-j\underline{k}'(2\pi\underline{N}^{-1})\underline{n}] . \quad (23)$$

Fast Fourier transform algorithms exist whenever  $\underline{N}$  is a composite matrix; that is, whenever  $\underline{N}$  can be factored into a nontrivial product of integer matrices. This is consistent with the existence condition for a 1-D FFT, which requires that the length of the 1-D DFT be a composite integer. As in the 1-D case, we shall see that the more factors that can be found for  $\underline{N}$ , the greater the computational savings.

### 3.1 Factorization of $\underline{N}$

In the remainder of this section whenever we refer to a matrix, it should be understood that we are referring to a matrix with integer elements. For such matrices  $|\underline{N}| = |\det \underline{N}|$  must be an integer. We have already commented upon the fact that this integer is equal to the number of samples in  $I_{\underline{N}}$ , or in one period of a periodic sequence with periodicity matrix  $\underline{N}$ .

Any matrix  $\underline{E}$  for which  $|\underline{E}| = 1$  is called a unit matrix.  $\underline{E}^{-1}$  is also a unit matrix. These are the only integer matrices whose inverses are integer matrices. If  $|\underline{N}|$  is a prime number, then we will say that  $\underline{N}$  is a prime matrix; if  $\underline{N}$  is neither a prime nor a unit matrix we will say that it is composite. It can be proved that any 2 x 2 composite matrix can be factored into a product of two matrices

$$\underline{N} = \underline{P}\underline{Q} \quad (24)$$

where neither  $\underline{P}$  nor  $\underline{Q}$  is a unit matrix. The 2 x 2 case includes all two-dimensional discrete Fourier transforms. It is undoubtedly true that all composite matrices can be factored, but the only proof known has not been generalized to larger integer matrices. It can be noted that the factorization in (24) is not unique since

$$\underline{N} = [\underline{P}\underline{E}][\underline{E}^{-1}\underline{Q}] \quad (25)$$

is also a factorization for any unit matrix  $\underline{E}$ , if  $\underline{P}$  and  $\underline{Q}$  are factors of  $\underline{N}$ . It should also be noted that the ordering of the factors may not commute.

The region of support,  $I_{\underline{N}}$ , of the sequence  $x(\underline{n})$  and the periodicity matrix,  $\underline{N}$ , of the transform must be consistent, i.e.,  $I_{\underline{N}}$  must denote one period of a periodic sequence with periodicity matrix  $\underline{N}$ . For a given  $\underline{N}$ , however,  $I_{\underline{N}}$  is not unique. This creates some confusion, not only for the derivation but also for its ultimate implementation as a computer program -- a confusion which was present in the 1-D case as well. The DFT, as we have seen, is basically a representation for a periodic sequence; its properties are determined solely by  $\underline{N}$ . Since several possible choices for  $I_{\underline{N}}$  can yield the same  $\underline{N}$ , sequences defined over different regions of support will have the same DFT. The DFT is unique only after  $I_{\underline{N}}$  is specified.

We will say that two integer vectors  $\underline{m}$  and  $\underline{n}$  are congruent with respect to the matrix modulus  $\underline{N}$  if

$$\underline{m} = \underline{n} + \underline{N}\underline{r} \quad (26)$$

for some integer vector  $\underline{r}$ . Each sample of a periodic sequence is congruent to the equivalent samples on other periods of the sequence. Thus, every sample of  $\tilde{x}(\underline{n})$  is congruent to a sample from  $I_{\underline{N}}$ . We shall use the notation

$$\underline{n} = ((\underline{n}))_N \quad (27)$$

to denote that sample in  $I_N$  which is congruent to  $\underline{n}$ .

### 3.2 Decomposition of $\underline{n}$ and $\underline{k}$

Any vector  $\underline{n}$  in the region  $I_N$  can be uniquely expressed as

$$\underline{n} = ((\underline{Pq} + \underline{p}))_N \quad (28)$$

where  $\underline{p} \in I_P$  and  $\underline{q} \in I_Q$ . The set of points  $I_P$  contains  $|P|$  integer vectors and the set  $I_Q$  contains  $|Q|$  integer vectors. We will have more to say about the exact composition of these sets in the next section. Here it is sufficient to know only of their existence. Furthermore any pair of vectors -- one from  $I_P$  and one from  $I_Q$  -- determine a unique element from  $I_N$ . In a crude sense the vector  $\underline{q}$  can be interpreted as the "quotient" when  $\underline{n}$  is "divided" by  $\underline{P}$  and  $\underline{p}$  can be interpreted as the "remainder."

In a similar fashion we can define

$$\underline{k}' = \underline{\ell}' + \underline{m}'Q \quad (29)$$

where  $\underline{m} \in J_P$  and  $\underline{\ell} \in J_Q$ . The regions  $J_P$  and  $J_Q$  are two sets whose sum spans  $J_N$ ; they contain  $|P|$  and  $|Q|$  samples respectively. With these definitions, the DFT sum in (23) can be rewritten as

$$X(\underline{Q}'\underline{m}+\underline{\ell}) = \sum_{\underline{p} \in I_P} \sum_{\underline{q} \in I_Q} x((\underline{Pq}+\underline{p}))_N \exp[-j(\underline{\ell}'+\underline{m}'Q)(2\pi N^{-1})(\underline{Pq}+\underline{p})] \quad (30)$$

which, through the exploitation of (24), can be expanded to

$$X(\underline{Q}'\underline{m}+\underline{\ell}) = \sum_{\underline{p} \in I_P} \exp[-j\underline{\ell}'(2\pi N^{-1})\underline{p}] \exp[-j\underline{m}'(2\pi P^{-1})\underline{p}] \sum_{\underline{q} \in I_Q} x((\underline{Pq}+\underline{p})) \exp[-j\underline{\ell}'(2\pi Q^{-1})\underline{q}] \quad (31)$$

This equation represents the first level of decomposition of a decimation-in-time Cooley-Tukey FFT algorithm. If  $\underline{PQ} = \underline{QP}$  a decimation-in-frequency form of the algorithm can be derived through the substitutions

$$\underline{n} = \underline{QP} + \underline{q} \quad (32a)$$

$$\underline{k}' = \underline{\ell}'\underline{P} + \underline{m}' \quad (32b)$$

T  
first  
quence  
This f

Thus t  
respec  
the re  
vector  
Q can

T  
bined  
denote  
(somet  
matrix  
multip  
plicall  
or Q is

3.3 T  
—  
A  
region

form a  
sampli

for a l  
region  
of q w  
congrue  
Th  
determi  
consist

To understand (31), it is appropriate that we look at it in pieces. Consider first the summation over  $\underline{q}$ . The sequence  $x((\underline{P}\underline{q} + \underline{p}))_{\underline{N}}$  when interpreted as a sequence defined over the vector variable  $\underline{q}$  is periodic with periodicity matrix  $\underline{Q}$ . This follows since

$$((\underline{P}(\underline{q}+\underline{Q})+\underline{p}))_{\underline{N}} = ((\underline{P}\underline{q}+\underline{p}+\underline{P}\underline{Q}))_{\underline{N}} = ((\underline{P}\underline{q}+\underline{p}))_{\underline{N}}$$

Thus the sum over  $\underline{q}$  in (31) represents a 2-D DFT of the array  $x((\underline{P}\underline{q}+\underline{p}))_{\underline{N}}$  taken with respect to the periodicity matrix  $\underline{Q}$ . The region of support for this sequence is the region  $I_{\underline{Q}}$ . A different matrix- $\underline{Q}$  DFT must be evaluated for each value of the vector  $\underline{p}$ . This means that  $|\underline{P}|$  such transforms need to be evaluated. If the matrix  $\underline{Q}$  can be factored into non-trivial factors, then the decomposition can be continued.

The summation on  $\underline{p}$  shows how the outputs of these matrix- $\underline{Q}$  DFT's should be combined to produce the matrix- $\underline{N}$  DFT. Let the outputs of the  $\underline{p}$ -th matrix  $\underline{Q}$  DFT be denoted as  $C(\underline{l}, \underline{p})$ . These numbers are multiplied by the factors  $\exp[-j\underline{l}'(2\pi\underline{N}^{-1})\underline{p}]$  (sometimes called twiddle factors) and the products are combined in a series of matrix- $\underline{P}$  DFT's, which are sometimes called butterflies. The number of twiddle factor multiplications is  $|\underline{N}|$  and the number of matrix- $\underline{P}$  DFT's or butterflies is  $|\underline{Q}|$ . Typically  $|\underline{P}|$  is small and  $|\underline{Q}|$  is large, but such need not be the case. If either  $\underline{P}$  or  $\underline{Q}$  is composite, either set of smaller DFT's can be further decomposed.

### 3.3 The Regions $I_{\underline{P}}$ , $I_{\underline{Q}}$ , $J_{\underline{P}}$ , and $J_{\underline{Q}}$

At this point it is appropriate to give some consideration to the form of the regions  $I_{\underline{P}}$ ,  $I_{\underline{Q}}$ ,  $J_{\underline{P}}$ , and  $J_{\underline{Q}}$ . These were defined by eqs. (28) and (29). The points

$$\underline{n} = ((\underline{P}\underline{q}))_{\underline{N}} \quad (33)$$

form a subset of the samples in  $I_{\underline{N}}$ , which are formed by sampling  $I_{\underline{N}}$  with the sampling matrix  $\underline{P}$ . The samples

$$\underline{n} = ((\underline{P}\underline{q} + \underline{p}))_{\underline{N}} \quad (34)$$

for a fixed value of  $\underline{p}$  represent a coset with respect to this sample set. The region  $I_{\underline{P}}$  consists of the coset leaders and the set  $I_{\underline{Q}}$  consists of a set of values of  $\underline{q}$  which will generate the subset defined by (33). The members of any coset are congruent with respect to the modulus  $|\underline{P}|$ .

The regions  $J_{\underline{P}}$  and  $J_{\underline{Q}}$  can be formed similarly. Whereas the region  $I_{\underline{N}}$  was determined to be consistent with the periodicity matrix  $\underline{N}$ , the region  $J_{\underline{N}}$  must be consistent with the frequency domain periodicity matrix  $\underline{N}'$ . (The transpose of  $\underline{N}$ ).

Since

$$\begin{aligned}\underline{N}' &= \underline{Q}'\underline{P}' \\ \underline{k} &= \underline{Q}'\underline{m} + \underline{l}\end{aligned}\tag{35}$$

we see that in the frequency domain  $\underline{Q}'$  plays a role which is similar to  $\underline{P}$  in the spatial domain, and  $\underline{P}'$  plays a role which is similar to  $\underline{Q}$ . Specifically  $\underline{Q}'$  is the frequency domain sampling matrix. Thus samples of the form

$$\underline{k} = ((\underline{Q}'\underline{m}))_{\underline{N}'}\tag{36}$$

form a subset of samples from the frequency domain. The region  $J_{\underline{P}}$  should consist of a set of  $|\underline{P}|$  vectors which will generate that subset. The region  $J_{\underline{Q}}$  will consist of a set of  $|\underline{Q}|$  coset leaders which, when added to the vectors given by (36), will give the remaining samples of  $J_{\underline{N}}$ .

These points should be made clearer in the example which is presented in the next section.

#### 3.4 An Example

As an example to illustrate the general FFT algorithm, let us consider the evaluation of a 4 x 4 rectangular DFT which has the periodicity matrix

$$\underline{N} = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}\tag{37}$$

The factorization of  $\underline{N}$  that we will use is

$$\underline{P} = \begin{bmatrix} 2 & 2 \\ 1 & -1 \end{bmatrix}; \quad \underline{Q} = \begin{bmatrix} 1 & 2 \\ 1 & -2 \end{bmatrix}\tag{38}$$

In Figure 1, we show the region  $I_{\underline{N}}$ . Those samples from  $I_{\underline{N}}$  which are of the form  $\underline{n} = ((\underline{P}\underline{q}))_{\underline{N}}$  are shown as asterisks with the remaining samples shown as dots. One set of vectors,  $\underline{q}$ , which will span the subset indicated by the asterisks is

$$I_{\underline{Q}} = ((0,0)', (1,0)', (2,0)', (3,0)')\tag{39}$$

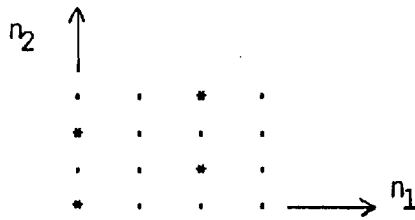


Figure 1 -- The region  $I_N$  for a 4 x 4 rectangular DFT. Those samples which are generated by sampling with  $\underline{P}$  in (38) are displayed as \*.

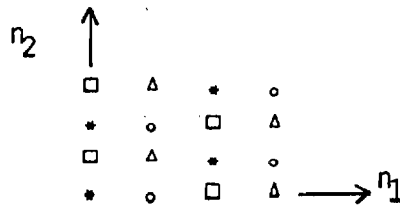


Figure 2 -- The four cosets that make-up  $I_N$  for the FFT algorithm corresponding to the decomposition of eq. (38).

In Figure 2 we show the four cosets that result from this factorization of  $\underline{N}$ . Note that all four cosets have the same geometry when periodically extended. The set  $I_P$  must be chosen to consist of four vectors which will describe the relative displacements of the cosets from one another. One possible choice is

$$I_P = \{(0,0)', (1,0)', (2,0)', (3,0)'\} \quad (40)$$

In Figure 3 we show the four cosets defined by  $\underline{Q}'$  for the region  $J_N$ . By referring to this figure, we find that possible choices for  $J_P$  and  $J_Q$  are

$$J_P = \{(0,0)', (1,0)', (2,0)', (3,0)'\} \quad (41)$$

$$J_Q = \{(0,0)', (0,1)', (0,2)', (0,3)'\} \quad (42)$$

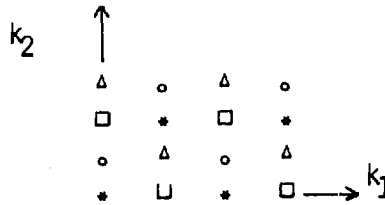


Figure 3 -- The four cosets that make-up  $J_N$  for the FFT algorithm corresponding to the decomposition of eq. (38).

Once  $I_P$ ,  $I_Q$ ,  $J_P$ , and  $J_Q$  have been chosen, a flowchart for the algorithm can be drawn from (31). This is done in Figs. 4-6. Figure 4 shows a matrix  $Q$  DFT; Figure 5 shows a matrix  $P$  butterfly; and Figure 6 shows how they are connected together. It should be noted that for this example the twiddle factors are all unity.

### 3.5 Computational Efficiency

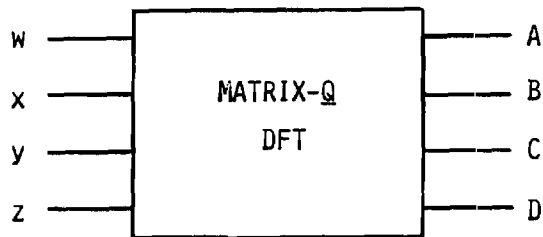
If  $C_N$  denotes the computational complexity of a matrix  $N$  FFT algorithm which is measured say in terms of the number of complex multiplications then

$$C_N \leq |P| C_Q + |Q| C_P + |N| . \quad (43)$$

The first term represents the number of multiplies in  $|P|$  matrix- $Q$  DFT's, the second term represents the contribution from the  $|Q|$  matrix- $P$  butterflies, and the last term represents the computation associated with the multiplications by the twiddle factors. This is shown as an upper bound because in some cases the number may be less. For example, in the example of the previous section there were no multiplies associated with the twiddle factors. Actually in that example no multiplies were required. If  $N$  is highly composite and the factors  $P_i$  are such that that  $|P_i| = 2, 4, 8, \text{ or } 16$ , often  $C_P$  will be zero. If

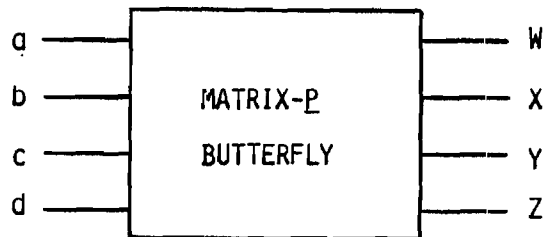
$$N = \prod_{i=1}^v P_i$$

then



$$\begin{aligned}
 A &= w + x + y + z \\
 B &= w - jx - y + jz \\
 C &= w - x + y - z \\
 D &= w + jx - y - jz
 \end{aligned}$$

Figure 4 -- A matrix-Q DFT for the example of section 3.4.



$$\begin{aligned}
 W &= a + b + c + d \\
 X &= a + jb - c - jd \\
 Y &= a - b + c - d \\
 Z &= a - jb - c + jd
 \end{aligned}$$

Figure 5 -- A matrix-P butterfly for the example of section 3.4.

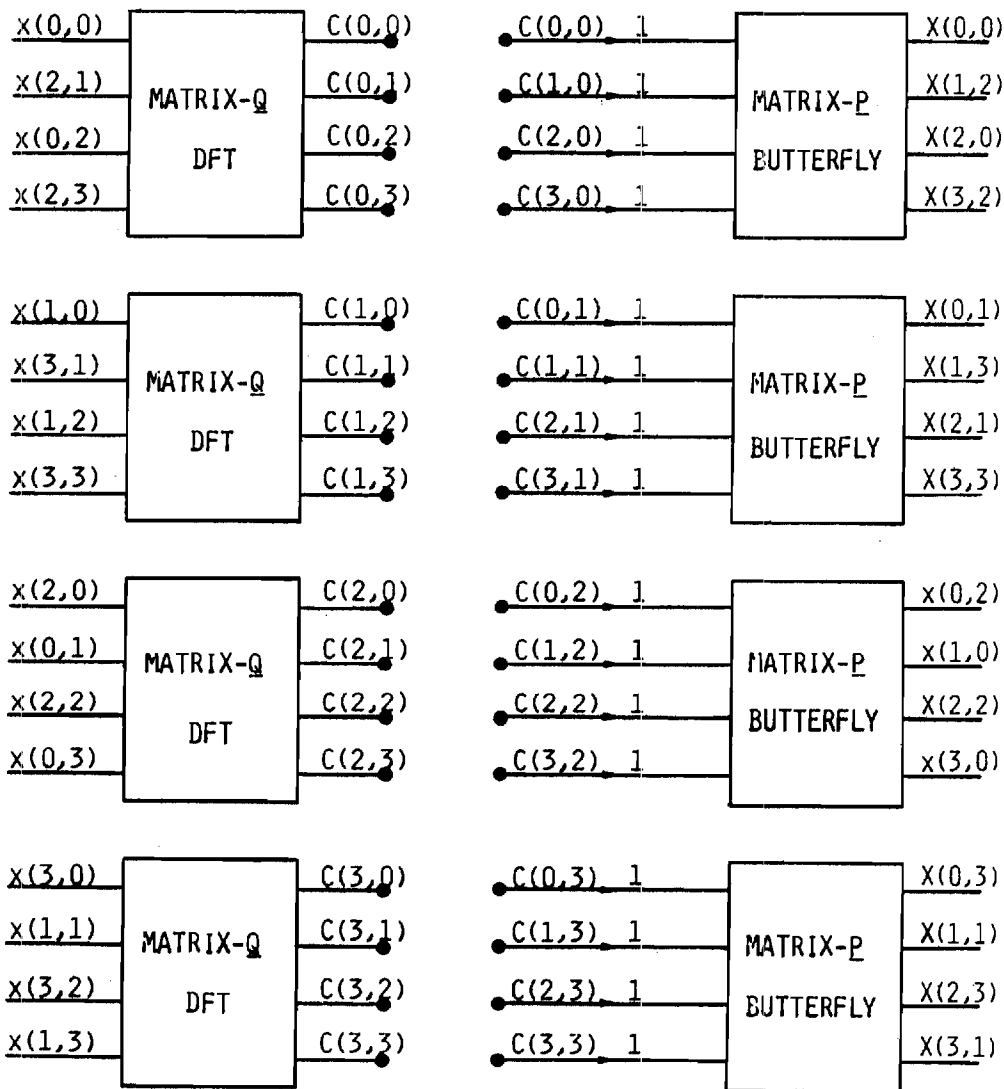


Figure 6 -- The complete flowchart of the 4 x 4 FFT for the example of section 3.4.

$$C_{\underline{N}} \leq |\underline{N}| \cdot \sum_{i=1}^v |\det \underline{P}_i| \quad (44)$$

but this bound tends to be conservative for many transforms of practical interest.

#### IV. RELATION OF THE GENERALIZED FFT TO STANDARD MULTI-DIMENSIONAL FFT ALGORITHMS

The two standard FFT algorithms for evaluating the DFT of rectangularly-sampled data are the row-column decomposition and the vector radix algorithm [1]. Both of these follow from the generalized algorithm for specific factorization the periodicity matrix,  $\underline{N}$ . The periodicity matrix of a 2-D rectangular DFT has the form

$$\underline{N} = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (45)$$

and the sets  $I_{\underline{N}}$  and  $J_{\underline{N}}$  are generally chosen to correspond to rectangularly-shaped regions of support;

$$I_{\underline{N}} = \{(n_1, n_2) : 0 \leq n_1 < N_1 \text{ and } 0 \leq n_2 < N_2\}$$

$$J_{\underline{N}} = \{(k_1, k_2) : 0 \leq k_1 < N_1 \text{ and } 0 \leq k_2 < N_2\}$$

The basic row column algorithm corresponds to one of the two factorizations

$$\underline{N} = \underline{P}_1 \underline{Q}_1 = \begin{bmatrix} N_1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (46)$$

or

$$\underline{N} = \underline{P}_2 \underline{Q}_2 = \begin{bmatrix} 1 & 0 \\ 0 & N_2 \end{bmatrix} \begin{bmatrix} N_1 & 0 \\ 0 & 1 \end{bmatrix} \quad (47)$$

With the factorization in (46) the column transforms are performed before the row transforms and with the factorization in (47) the row transforms are performed before the column transforms. With the factorization in (46) we can identify the sets

$$I_{\underline{P}} = \{(n_1, 0) : 0 \leq n_1 < N_1\}$$

$$I_Q = \{(0, n_2)'\}: 0 \leq n_2 < N_2\}$$

$$J_P = \{(k_1, 0)'\}: 0 \leq k_1 < N_1\}$$

$$J_Q = \{(0, k_2)'\}: 0 \leq k_2 < N_2\}$$

If  $N_1$  and  $N_2$  are each powers of 2, we can further decompose  $P$  and  $Q$  to get

$$\underline{N} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \cdots \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \cdots \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}. \quad (48)$$

This corresponds to the use of a 1-D radix-2 FFT to evaluate the column and row transforms.

If  $N_1$  and  $N_2$  are each divisible by two, it is also possible to perform the factorization

$$\begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} N_1/2 & 0 \\ 0 & N_2/2 \end{bmatrix}. \quad (49)$$

In this case the sets  $I_P$ ,  $I_Q$ ,  $J_P$ , and  $J_Q$  are

$$I_P = \{(0,0)', (0,1)', (1,0)', (1,1)'\}$$

$$I_Q = \{(n_1, n_2): 0 \leq n_1 < \frac{N_1}{2}, 0 \leq n_2 < \frac{N_2}{2}\}$$

$$J_P = \{(0,0)', (0,1)', (1,0)', (1,1)'\}$$

$$J_Q = \{(k_1, k_2): 0 \leq k_1 < \frac{N_1}{2}, 0 \leq k_2 < \frac{N_2}{2}\}$$

This factorization of  $\underline{N}$  corresponds to the first stage of decimation for a vector-radix algorithm. If  $N_1 = N_2 = 2^v$  the complete factorization for the radix- $(2 \times 2)$  FFT is

$$\underline{N} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \cdots \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}. \quad (50)$$

This approach can be readily applied to the cases of other radices, or to higher dimensional transforms.

In [7], Mersereau presented a discrete Fourier transform algorithm for hexagonally-sampled signals and a fast Fourier transform algorithm for evaluating it. That algorithm can be readily seen to correspond to the factorization

$$\underline{N} = \begin{bmatrix} 2N & N \\ N & 2N \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} N & N/2 \\ N/2 & N \end{bmatrix} \quad (51)$$

which leads to a vector-radix type algorithm. The regions  $I_P$  and  $I_Q$  for the first stage of the algorithm are

$$I_P = J_P = \{(0,0)', (0,1)', (1,0)', (1,1)'\}$$

$$I_Q = J_Q = \{(q_1, q_2) : 0 \leq q_1 < \frac{3N}{2}, 0 \leq q_2 < \frac{N}{2}\}.$$

If  $N$  is a power of two we can get the more complete decomposition

$$\underline{N} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \dots \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}.$$

The butterflies for all of the stages except the first are alike and contain four inputs and four outputs; those in the first stage contain three inputs and three outputs.

An alternative factorization for this transform is

$$N = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} N & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & N \end{bmatrix}$$

which provides a row-column algorithm for the hexagonal DFT.

## V. SUMMARY

In this paper we have presented a very general approach to understanding discrete Fourier transforms of multi-dimensional data. The key to that understanding is the role of the periodicity matrix. In essence once the periodicity matrix is specified the transform is defined. This general formulation incorporates many general DFT's where different sampling strategies are allowed in the spatial and Fourier domains.

We then showed that Cooley-Tukey-like fast Fourier transform algorithms could be developed to evaluate these general DFT's whenever the periodicity matrix could be factored into non-trivial factors and showed how several of the existing fast Fourier transform algorithms corresponded to specific factorizations of their periodicity matrices.

One of the nice features of this approach is that all of the information concerning the efficiency of an algorithm is contained in the periodicity matrix. This allows us the possibility to compare several algorithms and optimize them solely by examining these factorizations.

#### REFERENCES

- [1] D.B. Harris, J.H. McClellan, D.S.K. Chan, and H.W. Schuessler, "Vector radix fast Fourier transform," 1977 IEEE Int. Conf. on ASSP Record, pp. 548-551, 1977.
- [2] G.E. Rivard, "Direct fast Fourier transform of bivariate functions," IEEE Trans. Acoust. Speech, and Signal Processing, vol. ASSP-25, pp. 250-252, 1977.
- [3] E.A. Hoyer and W.R. Berry, "An algorithm for the two-dimensional FFT," 1977 IEEE Int. Conf. on ASSP Record, pp. 552-555, 1977.
- [4] B. Arambepola, "Fast computation of multi-dimensional discrete Fourier transforms," IEEE Proc., Vol. 127, pp. 49-52, 1980.
- [5] G.L. Anderson, "A stepwise approach to computing the multidimensional fast Fourier transform of large arrays," IEEE Trans. Acoust. Speech, and Signal Processing, Vol. ASSP-28, pp. 280-284, 1980.
- [6] J.W. Cooley and J.W. Tukey, "An algorithm for the machine calculation of complex Fourier series," Math. Comput., Vol. 19, pp. 296-301, 1965.
- [7] R.M. Mersereau, "The processing of hexagonally sampled two-dimensional signals," Proc. IEEE, Vol. 67, pp. 930-949, 1979.
- [8] R.M. Mersereau and D.E. Dudgeon, "The representation of two-dimensional sequences as one-dimensional sequences," IEEE Trans. Acoustics, Speech, and Signal Processing, Vol. ASSP-22, pp. 320-325, 1974.

FEEDBACK

#### ABSTRACT

This paper describes a near multiplexing 2-D system which, in effect, groups the problems into blocks. The excitation process on the input is already 2-D control

# OBJECTIVE QUALITY MEASURES FOR THE DESIGN OF DIGITAL IMAGE TRANSMISSION SYSTEMS

Bernd Girod

School of Electrical Engineering  
Georgia Institute of Technology  
Atlanta

## ABSTRACT

An analytical quality measure is essential for the successful design of digital transmission systems. For images the processes involved in human quality perception are too complicated to be modelled in a mathematically tractable way. In order to obtain useful results we have given up generality and investigated only quality measures for achromatic still images which have been coded by Pulse-Code-Modulation (PCM), Pseudo-Random Noise PCM, or Differential PCM. For this class of distortions we have found an objective quality measure based on the mean squared error, that performs better than commonly assumed. Another powerful measure has the same form as the mutual information between original and distorted image under the assumption of jointly Gaussian signals. In this paper these measures are compared with subjective results for the same distortions. The correlation is sufficiently good to justify the use of such measures in coding and evaluation of digital image coders.

## I. INTRODUCTION

The digitization of images introduces a distortion of the analog source signal. We know from rate-distortion theory that for a given information rate there exists a theoretical lower bound on this distortion [1]. An analytical distortion measure, or, equivalently, an analytical quality measure, is essential for computing this bound and for the design of a system that performs close to the theoretical optimum.

There are many further benefits that will result from having an objective fidelity criterion for images. Subjective tests are very time-consuming and expensive. Moreover, a meaningful comparison of subjective test results by different research groups is oftentimes not possible when different kinds of subjective tests are used. In contrast, numerical results of an objective measure are readily computed and allow a consistent comparison of different algorithms.

Our approach toward finding a satisfactory quality criterion consisted of three experimental stages. At first a data base of distorted images was generated. Then extensive tests yielded a subjective quality evaluation of these images by

human observers. Finally several objective distortion measures were computed and their performances with respect to the subjective results were compared.

The process from the initial optical input into the eye to the final decision about perceived quality is extremely complex, and we cannot expect to obtain a model that possesses both generality and mathematical tractability. Fortunately, most applications of a quality measure do not require generality since the distortions are a priori known to be of a certain kind. We restrict ourselves to the class of distortions that are caused when still achromatic images are encoded using Pulse-Code Modulation (PCM), Pseudo-Random Noise PCM (RNPCM), or Differential PCM (DPCM) [2,3,4].

A general simulation system produced from three originals "Girl" (G1), "Radome" (Rd), and "Hall" (Ha) the 161 distorted images as in Table I

Table I - Coding Distortion Data Base

	G1	Rd	Ha	Total
PCM	28	20	18	66
RNPCM	15	15	14	44
DPCM	14	22	15	51
Total	57	57	47	161

by varying the number of quantizer bits, quantizer range, number and sum of predictor coefficients, and companding function. The coding distortions included, in particular, contouring, granular noise, edge busyness, and slope overload [5].

## II. SUBJECTIVE IMAGE QUALITY TESTING

An isometric doubly-anchored test yielded the subjectively perceived qualities of the distorted images [6,7]. In this test the distorted images are shown simultaneously with two other images of given good and bad qualities and are rated on a scale 0 to 100. The anchors keep the deviation of the results in repeated tests small. Barnwell and Mersereau [7] showed the significance of scores that they had obtained by a doubly-anchored isometric image quality test by means of the Newman-Keul test [8]. As anchors we provided the original image with quality 80 and a heavily distorted image with quality 20. The display facility was calibrated

such that the mapping from the representing numbers in the simulation system into the luminance of the screen was linear. The viewing angle for one pixel was 0.022 degrees. This implies a highest possible spatial frequency of approximately 30 cpd (cycles per degree). The human eye can detect frequencies up to 40 cpd in the fovea of the retina [5]. The images were shown to approximately 15 subjects each in time intervals that depended on the response time of the individual subjects of typically 10 to 30 seconds.

The results of the subjective tests showed that an individual approximately achieves a correlation coefficient  $r=93\%$  with the average ratings by a set of individuals. This figure can serve as a reference for the performance of quality measures.

### III. THE OBJECTIVE MEASURES

The system that we used to test several objective measures is shown in Fig. 1. The Linear Quality Estimator (LQE) computes from  $L$  quality features  $F_1, F_2, \dots, F_L$  the best linear fit  $G$  to the subjective qualities  $Q$ . We can compare the quality of different objective quality measures by looking at the mean squared approximation error between  $G$  and  $Q$  or by estimating the correlation coefficient  $r$  between  $G$  and  $Q$  [7].

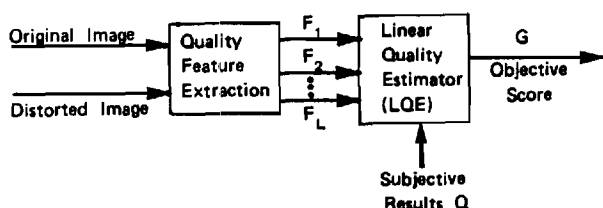


Figure 1 - System Used to Test Objective Measures

It must be borne in mind that the use of the LQE in Figure 1 implies the danger of overspecifying an objective measure. If we choose the number of quality features  $L$  to be too large, the quality estimator that has been optimized for one set of images  $X$  will very likely produce poor results and a small  $r$  for another set  $Y$ . We extend our notation for the correlation coefficient to  $r(X \rightarrow Y)$ , where " $X \rightarrow Y$ " means that the LQE has been specified for set  $X$  and that the correlation coefficient  $r$  has been computed over set  $Y$ . With this notation we operationally define the robustness of an objective quality measure

$$\rho(X, Y) = \frac{1 - \frac{1}{2} (|r(X \rightarrow X)| + |r(Y \rightarrow Y)| - |r(X \rightarrow Y)| - |r(Y \rightarrow X)|)}{1 - |r(X \rightarrow Y)| - |r(Y \rightarrow X)|} \quad (1)$$

For an ideally robust measure it is  $\rho(X, Y) = 1$ . Since the correlation coefficient  $r$  is invariant to linear transformations, any quality measure that uses only one nonconstant quality feature is ideally robust. The worst case is  $\rho(X, Y) = 0$  which implies a measure that can be tailored for one set but does not have any practical relevance for another set at all.

The task of finding a "good" objective quality measure can now be reformulated as finding an objective quality measure  $G$  with the high

correlation  $r$  to the subjective results  $Q$  under the constraint that its robustness is larger than some minimum number.

### 3.1 The Mean Squared Error Measure

Let us denote the samples that represent the original image as  $S(m, n)$ , and the distorted image as  $T(m, n)$ ,  $m=1, 2, \dots, M$ ,  $n=1, 2, \dots, N$ .  $S$  and  $T$  are digitized on an 8-bit integer scale from 50 to 305. We can define the mean squared error between  $S$  and  $T$  as

$$MSE \triangleq \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N (S(m, n) - T(m, n))^2 \quad (2)$$

$$\triangleq E((S-T)^2)$$

This measure is widely used for the design and optimization of image coding or transmission systems because of its simplicity and its excellent mathematical tractability. However, there always have been doubts concerning whether the MSE actually reflects human quality perception, and if it does, which classes of distortions can be so represented [5].

Using the MSE as a quality feature we computed the figures in Table IIa. Clearly, the robustness  $\rho=1$ , since only one feature was used. Figure 2a suggests that small MSE values have not been weighted heavily enough to cause severe deterioration of the objective quality, while the distortions with large MSE values have too much of an impact. In order to compensate for this effect, we apply a logarithmic nonlinearity to the MSE value and define the logarithmic mean squared error as

$$LMSE = \ln(MSE) \quad (3)$$

Its performance as a quality measure is shown in Table IIb and Fig. 2b. The correlation with subjective results comes close to the correlation that an individual human observer would achieve.

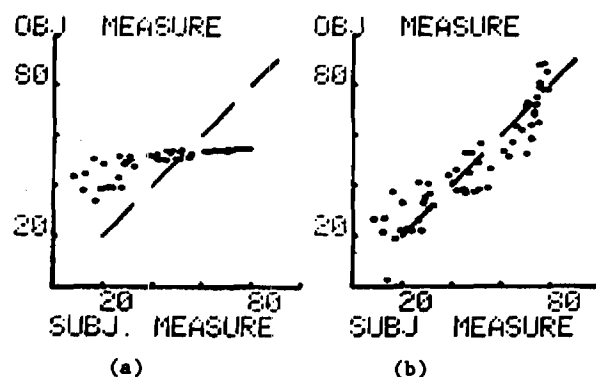


Figure 2 - Scatter Plot of Objective versus Subjective Quality for (a) the MSE (b) the LMSE Measure (G1)

In the optimization of a system it is equivalent to minimize either the MSE or the LMSE. This means that for our special class of distortions the MSE is a better quality indicator than it has been commonly assumed.

Suppose we present two identical images for comparison by a quality measure based on the LMSE. The result would be a quality  $G=\infty$ . Consequently we have to introduce a threshold  $\Theta_{MSE}$  such that, if the MSE is less than this threshold, no deterioration of the quality is perceived. If almost all distorted images are above threshold, i.e., their MSE is greater than  $\Theta_{MSE}$ , an estimate for  $\Theta_{MSE}$  can be derived from the optimum coefficient vector  $a$  of the LQE with

$$a_1 + a_2 \ln(\Theta_{MSE}) = 80 \quad (4)$$

This equation must hold for our measure to be continuous at the point  $MSE = \Theta_{MSE}$ . We obtain for the "Girl"-image  $\Theta_{MSE} = 11.6$ .

The existence of a threshold is not surprising. Considerable research effort has been devoted in the past to determine distortions that the eye does not perceive. This range is most important for the design of high quality systems.

Table II - Performance of Different Objective Measures

Measure	correlation r	robustness $\rho$ (Gl, Rd)
(a) MSE	.63	1
(b) LMSE	.90	1
(c) LMSE <sub>d</sub>	.91	1
(d) 4x4 Error PS	.88	.59
(e) 4x4 Error log PS	.95	.91
(f) Octave Bands	.93	.83
(g) LOB	.95	.97
(h) MI	.82	1
(i) LMI	.88	1
(j) LMI/LMSE <sub>d</sub>	.93	.994
(k) Human Subject	.93	--

The measures proposed so far do not account for the roughly logarithmic characteristic of the human retina [9]. We define

$$\nu = \ln(S) \quad \mu = \ln(T) \quad (5)$$

as the subjectively perceived brightness of the original and the distorted images. The performance of the logarithmic mean squared error on a subjective brightness scale

$$LMSE_d = \ln(E((\nu - \mu)^2)) \quad (6)$$

is shown in Table II c). From eq.(4) we again can compute a threshold  $\Theta_{MSE_d} = 6.7 \cdot 10^{-3}$ . The nonlinearity according to eq.(5) does not have a major impact on the performance of the quality measure for our experiment. Since the nonlinearity becomes increasingly important for images with higher dynamic range, all further measures proposed were computed on the subjective brightness scale.

### 3.2 Frequency-weighted Measures

The most commonly accepted model for the first stage of the human visual system consists of an initial pointwise nonlinearity cascaded with a bank of linear spatial filters that are tuned to certain frequencies [6,7,10,11,12,13]. The sensitivity of these channels is maximum at approximately 5 cpd.

We used the independent 10 of 4x4 equally spaced samples of the error power spectrum on a subjective scale as feature vector. This measure is not robust as shown in Table II d). Using the logarithm of the power spectrum samples yields a greatly improved measure (Table II e)).

The weakest point of the frequency-weighted measure just discussed is the even spacing of its power spectrum samples. This was done for computational convenience. However, there is evidence that the frequency channels of the human visual system are spaced in such a fashion that in the low frequency range the spacing is denser than in the high frequency range [11,13]. We took this fact into account and accumulated the error power in eight circularly symmetric octave frequency bands. Table II f) and g) show the results both for using the power in the octave bands as quality features and taking the logarithm of the power in the individual bands. The logarithmic octave band (LOB) approach is very promising, even though it does not yield a completely satisfactory robustness for the distortions that we are considering in this paper. Most error spectra of these distortions are essentially flat. In terms of approximating Q by G in Fig. 1 using the Linear Quality Estimator, this means that the components of the feature vector F are highly correlated and no reliable estimates for the weighting vector can be obtained. The frequency-weighted measures do not yield any significant improvement as compared to the LMSE<sub>d</sub> measure for our class of distortions.

### 3.3 The MI-Measure

The following measure has been called the MI-measure since its computation has been motivated by the mutual information rate between two jointly Gaussian processes [1]. We define

$$MI = - \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \ln \left( 1 - \frac{|\Phi_{\nu\mu}(\omega_1, \omega_2)|^2}{\Phi_{\nu}(\omega_1, \omega_2) \Phi_{\mu}(\omega_1, \omega_2)} \right) d\omega_1 d\omega_2 \quad (7)$$

$\Phi_{\nu\mu}(\omega_1, \omega_2)$  is the cross spectrum between  $\nu$  and  $\mu$ .  $\Phi_{\nu}(\omega_1, \omega_2)$  and  $\Phi_{\mu}(\omega_1, \omega_2)$  are the power spectra of  $\nu$  and  $\mu$ .  $\omega_1$  and  $\omega_2$  represent the two dimensions of digital frequency. The performances of MI and of

$$LMI = \ln(MI) \quad (8)$$

as quality features is shown in Table II h) and i).

We suspected that the LMSE and the LMI measure different aspects of the distortions and combined

them to a LMI/LMSE compound measure. This measure performs about as well as a single human observer would while still having an excellent robustness  $\rho = .994$  (Table IIj, Fig. 3). It correlates well enough with subjective results to justify its application for the design of, for example, a DPCM system.

#### IV. CONCLUSIONS

The mean squared error is a much better quality indicator than it has been commonly assumed for distortions induced by PCM, RNPCM, and DPCM. The subjectively perceived quality is a logarithmic function of the mean squared error. A lower threshold for this mean squared error exists below which no quality deterioration can be perceived. This threshold can be estimated from the quality perception above threshold. Future research should compare these estimates with direct measurements of the threshold.

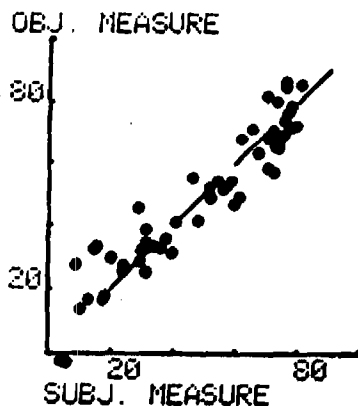


Figure 3 - Scatter Plot of Objective versus Subjective Quality for the LMI/LMSE Measure

Frequency-weighted measures are not desirable for systems which produce an essentially flat error power spectrum. However, note that the results of the LOB measure, a linear combination of the logarithm of the error power in octave bands, is very promising and should be investigated further in particular with respect to the individual thresholds in separated channels.

The MI-measure is interesting since it is a potential link to the constructs of rate-distortion theory.

The problems in the area of human image quality perception modelling are still far from being solved. Their understanding will be a key issue for the design of more effective coding schemes.

#### ACKNOWLEDGEMENTS

All results in this paper are part of a thesis [14] at Ga. Tech. I wish to thank Dr. R.M. Mersereau and Dr. R.W. Schafer for many helpful suggestions. This work was supported in part by the National Science Foundation under grant ECS 78-17201 and in part by the John and Mary Franklin Foundation.

#### REFERENCES

- [1] T. Berger, "Rate Distortion Theory," Englewood Cliffs: Prentice-Hall, Inc., 1971.
- [2] B.M. Oliver, J.R. Pierce, and C.E. Shannon, "The Philosophy of PCM," *Proc. IEEE*, vol. 36, pp.1324-1331, Nov. 1948.
- [3] L.J. Roberts, "Picture Coding Using Pseudo-Random Noise," *IRE Trans. Inform. Th.*, vol. IT-8, pp.145-154, Febr. 1962.
- [4] J.B. O'Neal, "Predictive Quantizing Systems (Differential Pulse Code Modulation) for the Transmission of Television Signals," *Bell Sys. Tech. J.*, pp.689-721, May-June 1966.
- [5] A.N. Netravali, J.O. Limb, "Picture Coding: A Review," *Proc. IEEE*, vol. 68, No. 3, pp.366-406, March 1980.
- [6] T.P. Barnwell, III., R.M. Mersereau, "A Comparison of Some Subjective and Objective Measures for Image Quality," *Eleventh Annual Asilomar Conf. on Circuits, Systems, and Computers*, pp.72-76, 1977.
- [7] T.P. Barnwell, III., B.M. Leiner, R.M. Mersereau, "Image encoding Subject to a Fidelity Measure," *Final Report E21-659 Grant ENG 75-04992 from National Science Foundation*, March 1978.
- [8] S.S. Wilks, "Mathematical Statistics," New York: J.Wiley and Sons, 1962.
- [9] H.R. Blackwell, "Contrast Thresholds of the Human Eye," *J. Opt. Soc. Am.*, vol. 36, pp.624-643, 1946.
- [10] H. Mostafavi, D. Sakrison, "Structure and Properties of a Single Channel in the Human Visual System," *Vision Research*, vol. 16, pp.957-968, 1976.
- [11] M.B. Sachs et. al., "Spatial Frequency Channels in Human Vision," *J. Opt. Soc. Am.*, vol. 61, pp.1176-1186, Sept. 1971.
- [12] C.F. Stromeyer, III., B. Julesz, "Spatial Frequency Masking in Vision, Critical Bands and Spread of Masking," *J. Opt. Soc. Am.*, vol. 61, pp.1175-1186, Sept. 1971.
- [13] W.A. Pearlman, "A Visual System Model and a New Distortion Measure in the Context of Image Processing," *J. Opt. Soc. Am.*, vol. 68, pp.374-386, 1978.
- [14] B. Girod, "Objective Quality Measures for the Design of Digital Image Transmission Systems," *Thesis at Georgia Institute of Technology*, Atlanta, 1980.

## TWO-DIMENSIONAL HEXAGONAL DIGITAL RECURSIVE FILTERS\*

P.A. Ramamoorthy

Western New England College  
Springfield, MA 01119

### ABSTRACT

It is known that band-limited two-dimensional (2D) signals can be sampled and processed using different periodic sampling strategies, the most common and perhaps well known approach being rectangular sampling. Recently, it has been pointed out that sampling techniques such as hexagonal sampling can offer substantial savings in terms of sampling density and computation time as compared to rectangular sampling. In this paper, we examine in detail, the design of recursive systems for processing of such hexagonally sampled 2D signals. It is shown that a method previously developed by the author for the design of rectangular recursive filters can be used for hexagonal recursive filter design, as well. The method guarantees the stability of the filters designed. Examples of hexagonal recursive filters designed using this method are given.

### THEORY

#### 2D Signal Processing and 2D Filters

Signals are an important part of our day to day life. They are used to bear or convey information. For example, we come across signals in such diverse fields as speech communication, data communication, bio-medical engineering, acoustics, sonar and others. Signal processing is concerned with the processing or manipulation of these signals to modify or extract some pertinent information from them. The tremendous success in digital IC technology during the last decade has made possible processing of signals using digital

\*This work was done while the author was a visiting Assistant Professor with the School of Electrical Engineering, Georgia Institute of Technology, Atlanta, Georgia and was supported by the National Science Foundation under Grant ECS-7817201.

techniques. The availability and still promising new developments in faster and cheaper digital hardware has resulted in practical application of digital signal processing techniques to existing one-dimensional (that is signals that are functions of a single variable) analog signal processing techniques and also opened up a whole variety of new applications. 2D digital signal processing is one such area which has received wide attention.

Basically, a 2D analog or continuous signal  $x_a(s,t)$  is a function of two independent variables  $s$  and  $t$ , where,  $s$  and  $t$  can take values from  $-\infty$  to  $+\infty$ . A 2D digital signal  $x(m,n)$  is a function of two independent variables  $m$  and  $n$ , where  $m$  and  $n$  can only take integer values. A 2D digital signal, though not necessarily derived from an analog signal, can be considered as having been obtained from an analog signal by sampling. Thus, we could say

$$x(m,n) = x_a(mT_1, nT_2) \quad (1)$$

where in (1), we have assumed rectangular sampling and  $T_1$  and  $T_2$  are the horizontal and vertical sampling intervals. In Fig. 1 we have shown a rectangular sampling raster and it should be noted that  $x(m,n)$  is defined only at the locations indicated by dots.

We can process a 2D digital signal using linear spatial frequency filtering. Comes under this category is Linear Shift-Invariant (LSI) filters in which the input and output satisfy a linear constant coefficient difference equation of the form

$$\sum_{k,l \in S} a(k,l)x(m-k,n-l) = \sum_{k,l \in S} d(k,l)y(m-k,n-l) \quad (2)$$

where  $y(m,n)$  is the output sequence and  $S$  denotes a finite region of support for the coefficients  $a(m,n)$  and  $d(m,n)$ . When the coefficients  $d(m,n)$  ( $m = n \neq 0$ ) are nonzero, we have the infinite-impulse response (IIR) filters having a transfer function of the form

$$H(Z_1, Z_2) = \frac{A(Z_1, Z_2)}{D(Z_1, Z_2)} = \frac{\sum_{k,l \in S} a(k,l)Z_1^{-k}Z_2^{-l}}{\sum_{k,l \in S} d(k,l)Z_1^{-k}Z_2^{-l}} \quad (3)$$

The region of support S in equations (2) and (3) can be either a single quadrant or a non-symmetric half-plane as shown in Fig. 2 (or their rotations), leading to the classification of quarter-plane (QP) and non-symmetric half-plane (NSHP) digital IIR filters.

The frequency response of a 2D LSI filter is obtained as given below:

$$H(\omega_1, \omega_2) = H(Z_1 = \exp(j\omega_1), Z_2 = \exp(j\omega_2)) \quad (4)$$

It can be shown that the response of 2D LSI filters are periodic in the  $\omega_1$  and  $\omega_2$  plane with the basic period as  $-\pi \leq \omega_1 < \pi$  and  $-\pi \leq \omega_2 < \pi$ . The design of LSI filters is best carried out in the frequency domain. In the case of IIR filters (QP and NSHP), this involves the selection (through iterative means) of the coefficients  $a(m,n)$  and  $d(m,n)$  such that the filter frequency response approximates the frequency response specifications. In the IIR filter design, we are also faced with the constraint that the resulting filter is stable, which, in terms of the filter transfer function  $H(Z_1, Z_2)$  amounts to requiring that (for QP filters)

$$D(Z_1, Z_2) \neq 0 \text{ in } U^{-2} \equiv \{|Z_1| \geq 1, |Z_2| \geq 1\} \quad (5)$$

and for NSHP filters

$$\begin{aligned} D(Z_1, Z_2) \neq 0 \text{ in } U^{-2} &\equiv \{|Z_1| = 1, |Z_2| \geq 1\} \\ D(Z_1, \infty) \neq 0 &\text{ for } \{|Z_1| \geq 1\} \end{aligned} \quad (6)$$

These two conditions are rather complicated to test and also difficult to meet with when standard mathematical algorithms are used for the design procedure.

#### Hexagonal Sampling and Filters

When a 2D digital signal is derived from an analog signal, we can adopt periodic sampling strategies other than the rectangular sampling considered so far. The fact that we have choice in the selection of sampling strategies, suggests that there may be some other method which is superior to rectangular sampling. In fact, it has been pointed out that hexagonal sampling can offer substantial savings in terms of sampling density and computation time as compared to rectangular sampling [1]. Hexagonal sampling can be described by

$$(m,n) = x_a \left( \frac{2m-n}{2} T_1, nT_2 \right) \quad (7)$$

This corresponds to sampling  $x(s,t)$  at the locations indicated in Fig. 3. Except for the change in the sampling raster, most of the principles of rectangular sampling and filters apply to hexagonal case too. Thus, we can talk about IIR filters which can be described in the partial domain by (2) or in the frequency domain by the transfer function in (3) (Let us use the subscripts R and H to denote rectangular and hexagonal filters). However, the region of support for the coefficients  $a(m,n)$  and  $d(m,n)$

varies slightly. Here, we can have a) one-sixth plane recursive filters, b) one-third plane recursive filters (similar to rectangular QP filters) and c) half-plane recursive filters (similar to NSHP filters) (Fig. 4). The stability equations in (5) and (6) applies respectively to cases (b) and (c) above. The frequency response of a hexagonal LSI filter is obtained as follows:

$$H(\omega_1, \omega_2) = H(Z_1 = \exp(j\frac{2}{3}\omega_1), Z_2 = \exp(j\omega_2 - \frac{j\omega_1}{3})) \quad (8)$$

It can be noted that the frequency response of a hexagonal LSI system is hexagonally periodic. In this paper, we consider the design of 2D hexagonal IIR filters. The design is based on a method previously developed by the author for the design of rectangular recursive filters and guarantees the stability of the filters designed. For completeness sake, we first present a review of rectangularly sampled IIR filter design and then present our results on hexagonal IIR filters.

#### REVIEW OF RECTANGULAR IIR FILTER DESIGN PROCEDURE

Consider the stability equations (5) and (6) for QP and NSHP filters respectively. As we noted before, the coefficients  $d(m,n)$  of the denominator of the transfer function have to be so constrained that they are met. The general 2D filter stability problem is solved in this method by noting the similarities between the transfer functions of 2D IIR systems and the transfer functions of 2D analog networks. In particular, if we apply a double bilinear transformation given by  $Z_1 = (1+s_1)/(1-s_1)$  to equation (5) for QP filters, the condition in (5) reduces to

$$Q(s_1, s_2) \neq 0 \text{ for } [\text{Re } s_1 > 0, \text{Re } s_2 > 0] \quad (9)$$

and equation (6) for NSHP filters reduces to

$$Q(s_1, s_2) \neq 0 \text{ for } [\text{Re } s_1 = 0, \text{Re } s_2 > 0] \quad (10a)$$

$$Q(s_1, 1) = (1-s_1)^m Q_1(s_1); \quad Q_1 \text{ strictly Hurwitz} \quad (10b)$$

It is well known in 2-variable (2V) analog network theory, that the numerator or denominator of the driving point function of a 2V passive network has the property given in (9). Therefore, one can obtain, for example, the denominator polynomial  $Q(s_1, s_2, y_{k1})$  of a properly connected 2V passive network (one such network is shown in Fig. 5a, where  $y_{k1}$ 's are the only real variables) and use it in the design of QP IIR filters. That is, we can rewrite the transfer function  $H(Z_1, Z_2)$  of a rectangular QP IIR filter as

$$H(Z_1, Z_2) = \frac{N(Z_1, Z_2)}{Q(s_1, s_2, y_{k1}) (Z_1+1)^m (Z_2+1)^n} \quad s_1 = \frac{Z_1-1}{Z_1+1} \quad (11)$$

and use the coefficients  $y_{k1}$  and that of the numerator as the variables of optimization. Because of the property of the polynomial  $Q(s_1, s_2, y_{k1})$ , the filter will be stable for all values of  $y_{k1}$ , thereby eliminating the need for stability checking. This approach has been successfully used for designing filters of order 7 in each variable [2].

## REFERENCES

1. R.M. Mersereau, "The Processing of Hexagonally Sampled Two-dimensional Signals", Proc. IEEE, vol. 67, PP 930-949, June 1979.
2. P.A. Ramamoorthy and L.T. Bruton, "Design of Stable Two-dimensional Analog and Digital Filters with Applications in Image Processing," Intl. J. Circuit Theory and Applications, vol. 7, pp 229-246, April 1979.
3. P.A. Ramamoorthy and L.T. Bruton, "Design of Stable Symmetric and Non-symmetric Half-plane Digital Recursive Filters," Proc. Intl. Conf. ASSP, pp 40-43, April 1979.

Now, coming to the case of rectangular NSHP filters, it can be shown [3] that the denominator of a 2V network containing both positive and negative valued reactive elements in  $s_1$ , and only positive valued reactive elements in  $s_2$ , as shown in Fig. 5b, will have the property in (10a). Furthermore, by constraining the network element values properly, we can force the denominator polynomial to satisfy (10b). Therefore, we can proceed in a manner similar to that of the design of QP filters and avoid the stability problem completely [3].

## HEXAGONAL FILTER DESIGN

Let us denote by  $H_H(Z_1, Z_2)$  and  $H_R(Z_1, Z_2)$  the transfer functions of hexagonal and rectangular IIR filters respectively. If the same set of coefficients is used to design both a rectangular and a hexagonal filter, then from (4) and (8), the frequency responses of the filters would be related by  $H_H(\omega_1, \omega_2) = H_R(\omega_1, \omega_2 - \frac{1}{3}\omega_1)$ . It is possible to use a rectangular filter design algorithm to design a hexagonal filter if the ideal frequency response is first transformed. Thus to design a hexagonal filter which approximates the ideal response  $I_H(\omega_1, \omega_2)$ , one would first define  $I_R(\omega_1, \omega_2) = I_H(\frac{3}{2}\omega_1, \omega_2 + \omega_1/2)$ . [Examples of this transformation or prewarping as applied to two ideal responses are shown in Fig. 6]. Arrays  $a(m, n)$  and  $d(m, n)$  can then be determined by the rectangular filter design algorithm discussed before to approximate  $I_R(\omega_1, \omega_2)$ . Those same coefficients, when interpreted as the coefficients of a hexagonal filter will result in a hexagonal design for which the frequency response approximates  $I_H(\omega_1, \omega_2)$ . Furthermore, since our rectangular filter design algorithm guarantees the stability at each iteration, the hexagonal filters will always be stable.

## NUMERICAL EXAMPLE

In this example, we present numerical results on the design of a hexagonal IIR filter approximating a circularly symmetric lowpass magnitude characteristic give in Fig. 6a. We decided to approximate this characteristic using a 1/3 plane filter with a sixth-order transfer function expressed as the product of 3 second-order transfer functions. In Fig. 7, the actual magnitude response of the filter is given. It can be noted that we have obtained an excellent approximation to the ideal characteristic using this sixth-order transfer function (The maximum deviation inside the passband was 0.34 dB).

## CONCLUSION

In this paper, we examined in detail, the design of hexagonal IIR filters. It is shown that a method previously developed by the author for the design of rectangular IIR filters can be used efficiently for the design of hexagonal IIR filters using prewarping of the given specifications. An example of a sixth-order one-third plane filter is given.

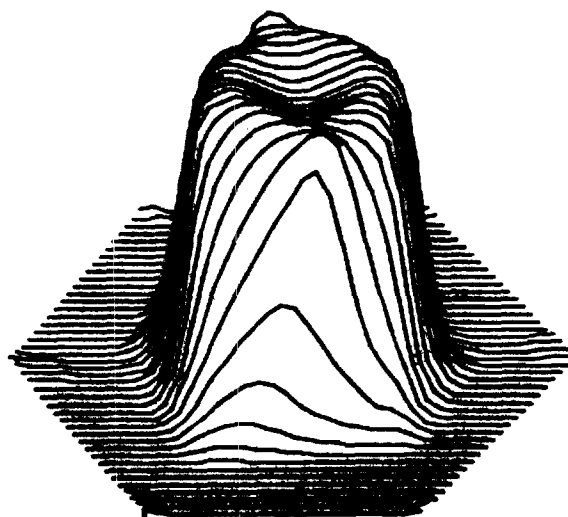


Fig. 7a 3D plot of the magnitude response of a sixth-order hexagonal IIR filter

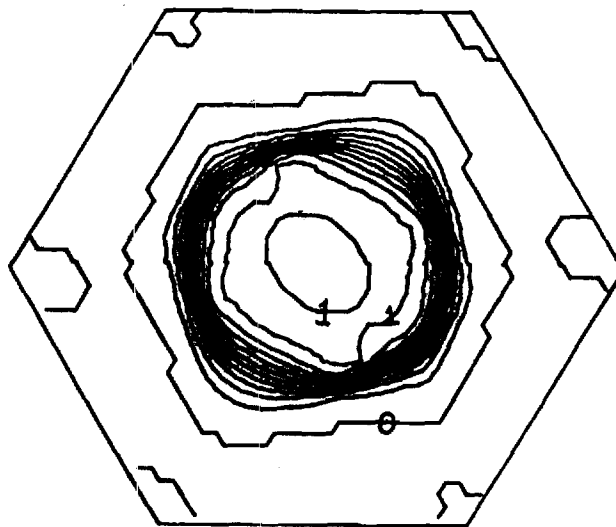


Fig. 7b Contour plot of the magnitude response.

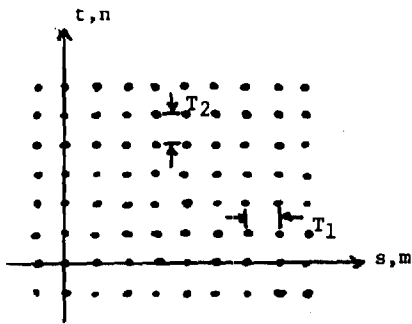


Fig. 1 A rectangular sampling raster

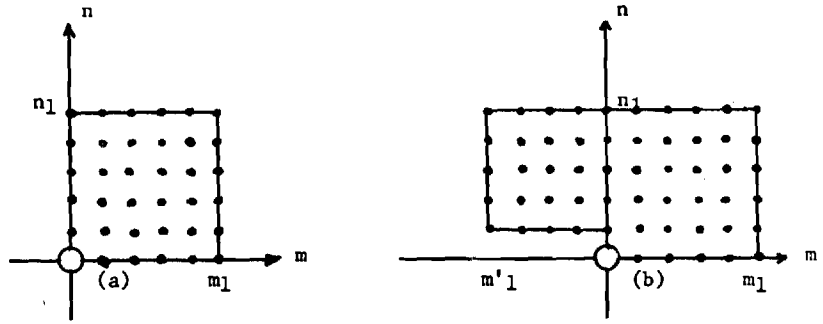


Fig. 2 Output masks corresponding to rectangular (a) QP recursive filters and (b) NSHP recursive filters

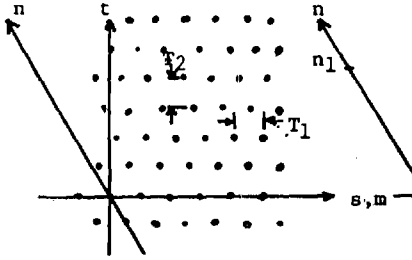


Fig. 3 A rectangular sampling raster

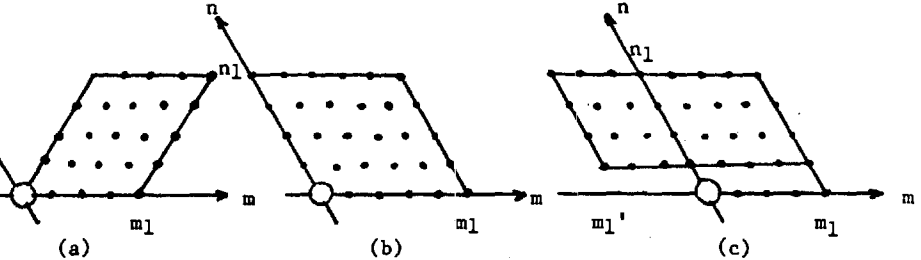


Fig. 4 Output masks corresponding to hexagonal (a) one-sixth plane recursive filters, (b) One-third plane recursive filters and (c) half-plane recursive filters.

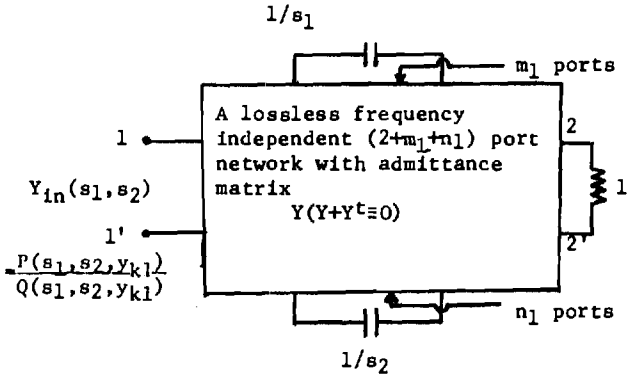
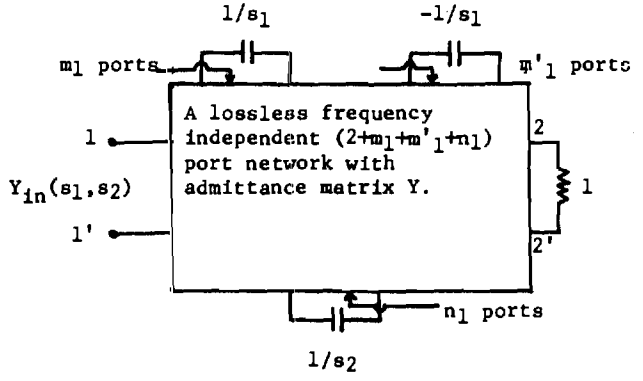


Fig. 5 A) A 2V passive network



B) A 2V network with negative valued elements.

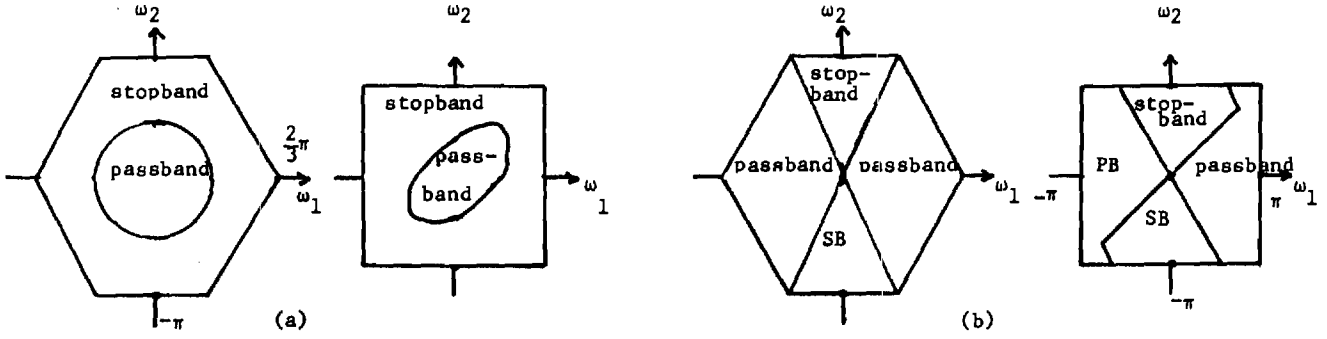


Fig. 6 Prewarping the frequency response from hexagonal filter specifications to rectangular filter specifications (a) Circularly-symmetric response, (b) Fan-filter response.

A SYSTEM FOR HELIUM SPEECH ENHANCEMENT USING  
THE SHORT-TIME FOURIER TRANSFORM\*

Mark A. Richards

School of Electrical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332

ABSTRACT

Helium speech is produced by speaking in a high-pressure helium-oxygen environment. A variety of acoustical and psychological effects render unprocessed helium speech virtually unintelligible. In this paper, a new system for enhancing helium speech based on the short-time Fourier transform (STFT) is described. The proposed system is able to correct many of the helium speech phenomena, does not require either pitch detection or a voiced/unvoiced decision, and is able to incorporate "spectral subtraction" noise reduction algorithms in a natural way. Alternative methods for estimating the spectral envelope from the STFT and correcting its distortion are described, as well as a technique for reducing the signal sampling rate at the earliest possible stage.

INTRODUCTION

A serious problem in deep sea diving is the physiological effects of breathing air at the high pressures required. Because the nitrogen in air leads to nitrogen narcosis and the "bends", a breathing gas consisting of helium and oxygen is commonly substituted. This technique solves the physiological problems but introduces a communications problem, namely that speech uttered in a high-pressure helium-oxygen atmosphere is virtually unintelligible. Consequently, there is a need to develop systems for enhancing helium speech so as to improve its intelligibility. Several systems have been proposed in the past for solving this problem.

The purpose of this paper is to describe a new system currently being developed for helium speech enhancement. This system is based upon a short-time Fourier transform (STFT) representation of the speech signal. It requires neither pitch detection nor a voiced/unvoiced decision capability. Considerable latitude is available in the tradeoff between the sophistication and the computational complexity of the enhancement operations. "Spectral subtraction" noise reduction techniques may be easily incorporated. Finally, in most cases the signal sampling rate can be halved at the earliest possible stage, thereby helping to minimize the computa-

tional load.

HELIUM SPEECH

Helium speech is characterized by a variety of distortions and degradations. The best understood and most important of these is the effect of the He-O<sub>2</sub> atmosphere on the spectral envelope of the speech signal. Measurements of formant frequencies in air and in a He-O<sub>2</sub> atmosphere show that the formants of helium speech are shifted upward in frequency; this shift has been described by the formula(1)

$$f_{he} = \sqrt{\alpha^2 f_a^2 + f_0^2} \quad (1)$$

In Eq. (1),  $f_a$  is the formant frequency for normal speech and  $f_{he}$  is the corresponding formant frequency for helium speech. The parameter  $\alpha$  is the ratio of the speed of sound in the helium atmosphere to that in air and is typically between 2 and 3. The constant  $f_0$ , which is dependent on both the gas composition and pressure, is typically between 150 and 400 Hz. Despite the formant frequency shift, the formant bandwidths remain approximately constant (2). The relationship between the spectral envelopes of normal and helium speech is therefore more complex than a simple warping of the frequency scale according to Eq. (1). Additionally, the natural rolloff of the speech spectrum is accentuated in helium speech; the combination of this effect and the spectral "stretching" results in very low levels for the upper formants of the helium speech signal.

A number of secondary effects are also found in helium speech. There may be an upward shift of the speaker's pitch by a factor of about 1.1 to 1.5 (3). There is a decrease in the energy of consonants relative to that of vowels (2). These effects are not as important as the spectral envelope distortion in determining intelligibility. Finally, the speaking environment usually has very poor acoustic qualities and is very noisy, due to the face masks and other breathing gear used by a diver.

The combination of all these factors makes helium speech enhancement a challenging problem.

\* This work was supported, in part, by the National Science Foundation under grant ECS-7817201.

Most systems which have been proposed for this task concentrate on correcting the spectral envelope warping. However, many such systems assume that the warping can be described by the linear asymptotic approximation to Eq. (1),  $f_{be} = \alpha f_a$ . Some other systems can perform a nonlinear warping but still only approximate Eq. (1). In any event, most systems compress the formant bandwidths when correcting the spectral warping. Furthermore, few systems have been proposed which are capable of pitch correction, or that incorporate noise reduction as an integral part of the algorithm. The STFT-based system proposed here attempts to overcome all of these common drawbacks. To see how this might be done, it is useful to consider a model for the short-time Fourier transform of speech.

#### THE NARROWBAND STFT OF SPEECH

The short-time Fourier transform of a signal  $x(n)$  is defined as

$$X(n, \omega) = \sum_{m=-\infty}^{\infty} x(m)h(n-m)e^{-j\omega m}. \quad (2)$$

The function  $h(n)$  is a sliding window which serves to isolate a portion of  $x(n)$  for analysis. A detailed discussion of the STFT is available in (4).

It is common to model speech production by a structure like that of Figure 1. Here a "slowly" time-varying filter  $t(n, m)$  representing the glottal pulse, vocal tract, and radiation effects is excited by either a quasi-periodic impulse train (for voiced speech) or a white random process (for unvoiced speech). Portnoff (4) has studied the STFT of speech modeled by this structure. For a sufficiently narrowband STFT (that is, where  $h(n)$  is relatively long compared to the local pitch period) he has shown that, in the voiced speech case,

$$X(n, \omega) \approx \begin{cases} a(n)T(n, \Omega(n))H(\Omega(n) - \omega)e^{-j\omega n}, & |\omega - \Omega(n)| < \omega_h \\ 0 & |\omega - \Omega(n)| > \omega_h \end{cases} \quad (3)$$

In Eq. (3),  $\Omega(n)$  is the local pitch frequency;  $T(n, \omega)$  is the Fourier transform (in  $m$ ) of  $t(n, m)$ ;  $H(\omega)$  is the Fourier transform of  $h(n)$  and  $\omega_h$  its approximate bandwidth; and  $a(n)$  is a complex number of constant magnitude but time-varying phase. Equation (4) verifies the well-known fact that the narrowband STFT of voiced speech is a series of lines at the pitch harmonics.

It is possible to further simplify the model for  $X(n, \omega)$ . Note that by setting  $T(n, \omega) = 1$  (equivalently,  $t(n, m) = \delta(m)$ ), the STFT of the voiced speech excitation function  $u(n)$  can be shown to be

$$U(n, \omega) \approx \begin{cases} a(n)H(\Omega(n) - \omega)e^{-j\omega n}, & |\omega - \Omega(n)| < \omega_h \\ 0 & |\omega - \Omega(n)| > \omega_h \end{cases} \quad (4)$$

If  $h(n)$  is long enough, then it can be assumed that  $T(n, \omega)$  is approximately constant in  $\omega$  over any interval of length  $\omega_h$ , so that  $T(n, \Omega(n))H(\Omega(n) - \omega) \approx T(n, \omega)H(\Omega(n) - \omega)$ . Equation (3) then becomes

$$X(n, \omega) \approx T(n, \omega)U(n, \omega). \quad (5)$$

The narrowband STFT of voiced speech can thus be modeled as a product of STFTs representing the source and the vocal tract. The source STFT  $U(n, \omega)$  is a series of evenly spaced spectral lines which carries the information on the speech pitch (in the line spacing) and on the analysis window  $h(n)$  (in the line shape).  $T(n, \omega)$  is the spectral envelope previously discussed.

For unvoiced speech, it can be shown that

$$E\{|X(n, \omega)|^2\} = E\{|U(n, \omega)|^2\} \{ |T(n, \omega)|^2 |H(\omega)|^2 \} \quad (6)$$

where  $E\{|U(n, \omega)|^2\}$  is just  $\sigma_u^2$ , the variance of  $u(n)$ . This is again a product of source and vocal tract terms, although in this case the spectral envelope is smoothed by  $|H(\omega)|^2$ . Equation (6) provides a rationale for applying the same processing to both voiced and unvoiced speech, obviating the need to distinguish the two cases. In the interest of brevity, only the voiced speech case is considered in the remainder of this paper.

#### THE ENHANCEMENT ALGORITHM

If the analysis window  $h(n)$  is properly chosen, the STFT of a helium speech signal  $x_{he}(n)$  can be assumed to be of the form of Eq. (5):

$$X_{he}(n, \omega) = T_{he}(n, \omega)U_{he}(n, \omega). \quad (7)$$

Since both the spectral envelope and the pitch will require correction in general, the STFT of the corresponding speech in air is assumed to take the form

$$X_a(n, \omega) = C(\omega)T_a(n, \omega)U_a(n, \omega). \quad (8)$$

A distinction has been drawn in Eq. (8) between correcting the frequency scale warping of  $T_{he}(n, \omega)$  and reversing the general attenuation of  $T_{he}(n, \omega)$  at higher frequencies.  $T_a(n, \omega)$  will be chosen to improve the spectral balance. The main steps of the helium speech enhancement algorithm are as follows:

1. Compute  $X_{he}(n, \omega)$  from  $x_{he}(n)$ . Assume that Eq. (7) holds.
2. Estimate  $T_{he}(n, \omega)$  from  $X_{he}(n, \omega)$  and

thereby decompose  $X_{he}(n, \omega)$  into its two components.

3. Estimate  $T(n, \omega)$  from  $T_{he}(n, \omega)$  so as to correct the spectral envelope distortion.
4. Estimate  $U_a(n, \omega)$  from  $U_{he}(n, \omega)$  so as to correct the pitch.
5. Select  $C(\omega)$  to improve the spectral balance.
6. Form the estimate of  $X_a(n, \omega)$  according to Eq. (8).
7. Compute  $x(n)$  from  $X_a(n, \omega)$  using standard methods (5).

Each of the operations 2 through 5 is further discussed below.

#### Estimating the Spectral Envelope

Four techniques have been considered for estimating  $T_{he}(n, \omega)$  from  $X_{he}(n, \omega)$ . The first three are essentially the same as considered by Makhoul (5), while the fourth is a new method. The linear predictive technique produces a minimum-phase estimate of  $T_{he}(n, \omega)$ , while the others estimate only  $|T_{he}(n, \omega)|$ .

Linear Predictive (LP) Method. In this method, the LP estimate of  $X_{he}(n, \omega)$  is computed and taken to be  $T_{he}(n, \omega)$ . The autocorrelation method has a computational advantage in this application (5). The required order of the LP estimate is typically 10 or 12, the same as for normal speech.

Homomorphic Method. This method uses the logarithm function to map  $|X_{he}(n, \omega)|$  into a space where the source and vocal tract terms are additive, so that linear filtering may be applied in an attempt to separate them. The log magnitude of  $X_{he}(n, \omega)$  is given by

$$\log|X_{he}(n, \omega)| = \log|T_{he}(n, \omega)| + \log|U_{he}(n, \omega)|. \quad (9)$$

Since  $T_{he}(n, \omega)$  is smooth compared to  $U_{he}(n, \omega)$ , lowpass filtering of  $\log|X_{he}(n, \omega)|$  in  $\omega$  yields an estimate of  $\log|T_{he}(n, \omega)|$ . Very short window functions (e.g., an 11-point Hamming window) make suitable lowpass filters and require little computation.

Smoothing Method. In this method,  $|X(n, \omega)|$  is smoothed by convolution with a suitable window. This approach is similar to the windowed autocorrelation technique of classical spectral analysis and also to ordinary envelope detection. Typical smoothing filters are the same as used for the homomorphic method.

Piecewise-Linear Method. This new technique applies a peak-picking algorithm to  $|X_{he}(n, \omega)|$  in order to locate the peaks of the individual spectral lines. It is necessary to devise constraints on the peak-picking process so that

nearly all spectral lines are identified without following the sidelobe ripples between the lines. Once found, the spectral peaks are joined by straight line segments to form a piecewise-linear approximation to  $|T_{he}(n, \omega)|$ .

Table I compares the operations involved in obtaining  $|T_{he}(n, \omega)|$  from  $|X_{he}(n, \omega)|^2$  for these four methods with typical parameter values. The piecewise-linear method requires the least computation, while the LP method requires by far the most. Informal listening indicates that all of the processes give results of similar quality. These observations would seem to indicate a clear preference for the piecewise-linear method. However, the LP technique may have an advantage when computing  $T_a(n, \omega)$  from  $T_{he}(n, \omega)$ . Also, even when using the LP method, the computation of Table I is only a portion of the total processing.

#### Correcting the Spectral Envelope

The simplest way to estimate  $T_a(n, \omega)$  from  $T_{he}(n, \omega)$  is by the relations

$$\xi(\omega) = \sqrt{\alpha^2 \omega^2 + \omega_0^2} \quad (10)$$

$$T_a(n, \omega) = T_{he}(n, \xi(\omega)). \quad (11)$$

The function  $\xi(\omega)$  is simply Eq. (1) recast into units of normalized radian frequency. This process corrects the formant frequencies but compresses their bandwidths. Two methods which address this problem are being considered for the case when the spectral envelope is estimated by the LP method. In this case the formant frequencies and bandwidths are determined by the angle and radius, respectively, of the roots of the predictor polynomial. It is thus possible to shift the formant frequencies while approximately maintaining their bandwidths by factoring the polynomial; rotating the roots in the z-plane while keeping their radii constant, so that the original angle is related to the new angle by Eq. (10); and then computing  $T_a(n, \omega)$  from the polynomial represented by these new roots.

The above method is accurate but time-consuming. A simpler but less accurate alternative is to multiply the predictor polynomial coefficient sequence by an exponential sequence  $r^{-k}$ . This scales the radii of the roots by the factor  $1/r$ ; if  $r > 1$ , the formant bandwidths are expanded. This procedure is used as pre-compensation prior to applying the scaling of Eqs. (10)-(11). A key problem with this method is the choice of  $r$ . The quality of the results obtained for  $T_a(n, \omega)$  by these various methods is under investigation.

#### Correcting Pitch Shift

The most straightforward way to correct an upward pitch shift by a factor  $\beta$  is by a linear scaling of  $U_{he}(n, \omega)$ :

$$U_a(n, \omega) = U_{he}(n, \beta\omega) \quad (12)$$

This reduces the line spacing, and so the implied pitch, by the factor  $\beta$ , yet does not require a pitch detection mechanism. However, this process also narrows the spectral lines by the factor  $\beta$ , implying in turn a lengthening of the effective window  $h(n)$  by that same factor. Consequently, the DFT size must be greater than the length of  $h(n)$  by a factor of at least  $\beta$  in order to avoid aliasing problems.

Correcting Spectral Balance

Although a number of factors contribute to the overall spectral imbalance of helium speech, the aggregate effect is primarily an attenuation of the upper reaches of the spectral envelope. Consequently, a simple 3 dB/octave emphasis is currently used to improve the spectral balance.

ADDITIONAL FEATURES

Noise Reduction by Spectral Subtraction

"Spectral subtraction" noise reduction algorithms (6) are based on an STFT signal representation and so are easily incorporated into the helium speech enhancement algorithm. It is first necessary to add a speech/silence decision capability. During "silent" intervals,  $|X_{he}(n, \omega)|^2$  is used to update a running average estimate of the noise power spectrum. During "speech" intervals the current noise spectrum estimate is subtracted from  $|X_{he}(n, \omega)|^2$  in order to suppress the noise components. This modified spectrum is used in the subsequent processing. The spectral subtraction technique, along with several embellishments used in the helium speech processor, is described in more detail in (6).

Reduction of Sampling Rate

Because  $\alpha$  in Eq.(1) is generally greater than 2, the estimate of  $X_a(n, \omega)$  will be band-limited to  $|\omega| < \pi/2$ . This implies that the sampling rate of  $x_a(n)$  may be reduced from that used to obtain  $x_{he}(n)$ . Suppose that  $x_{he}(t)$  is sampled at intervals of T seconds to obtain  $x_{he}(n)$  and that an N-point DFT is used to represent  $X_{he}(n, \omega)$ . The most direct way to obtain  $x_a(n)$  at a reduced sampling rate is to construct an N-point DFT representation of  $X_a(n, \omega)$ ; synthesize  $x_a(n)$  from this in the normal way, thereby obtaining a signal  $x_a(n)$  corresponding to a sampling interval T; and then apply conventional decimation techniques.

Suppose instead that only the DFT samples for  $k=0, \dots, (N/4)$  and  $k=(3N/4) + 1, \dots, N-1$  of  $X_a(n, \omega)$  are computed and that an  $(N/2)$ -point DFT representation for  $X_a(n, \omega)$  is constructed from these samples. It can be shown that the signal obtained by performing a conventional synthesis using the  $(N/2)$ -point DFT is  $x_a(2n)$ , indicating that the effective sampling rate has been halved. By halving the DFT size and the sampling rate prior to synthesizing  $x_a(n)$ , the computation required for the synthesis is reduced significantly.

References

1. R.F. Quick, "Helium Speech Translation Using Homomorphic Techniques", Air Force Cambridge Research Laboratories, Report AECRL-70-0424, July 1970.
2. T.A. Giordano, H.B. Rothman, and H. Hollien, "Helium Speech Unscramblers-A Critical Review of the State of the Art", IEEE Trans. on Audio and Electroacoustics, Vol. AU-21, pp. 436-444, October 1973.
3. J. Suzuki and M. Nakatsui, "Translation of Helium Speech by Splicing of Autocorrelation Function", J. Radio Research Laboratories (Japan), Vol. 23, No. 111, pp. 229-234, July 1976.
4. M.R. Portnoff, "Time-Scale Modification of Speech Based on Short-Time Fourier Analysis", Sc.D. Thesis, Mass. Institute of Technology, April 1978.
5. J. Makhoul, "Methods for Nonlinear Spectral Distortion of Speech Signals", Proceedings IEEE Int. Conf. on Acoustics, Speech and Signal Processing, pp. 87-90, April 1976.
6. M. Berouti, R. Schwartz, and J. Makhoul, "Enhancement of Speech Corrupted by Acoustic Noise" Proceedings IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 208-211, April 1979.

	LINEAR PREDICTIVE	SMOOTHING	HOMOMORPHIC	PIECEWISE LINEAR
MULT	24,244	1542	1542	405
DIV	257	0	0	150
ADD	12,468	1285	1285	762
LOG	0	0	257	0
EXP	0	0	257	0
SQR	521	0	0	0
SQ. RT	257	257	0	257

Table I  
Operations Required to Obtain  $|T_{he}(n, \omega)|$  from  $|X_{he}(n, \omega)|^2$

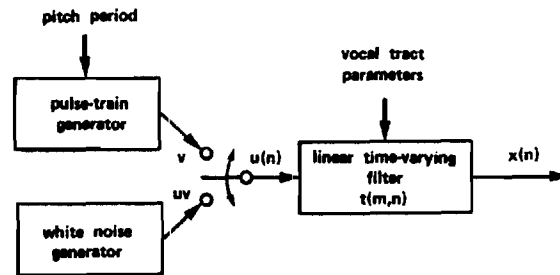
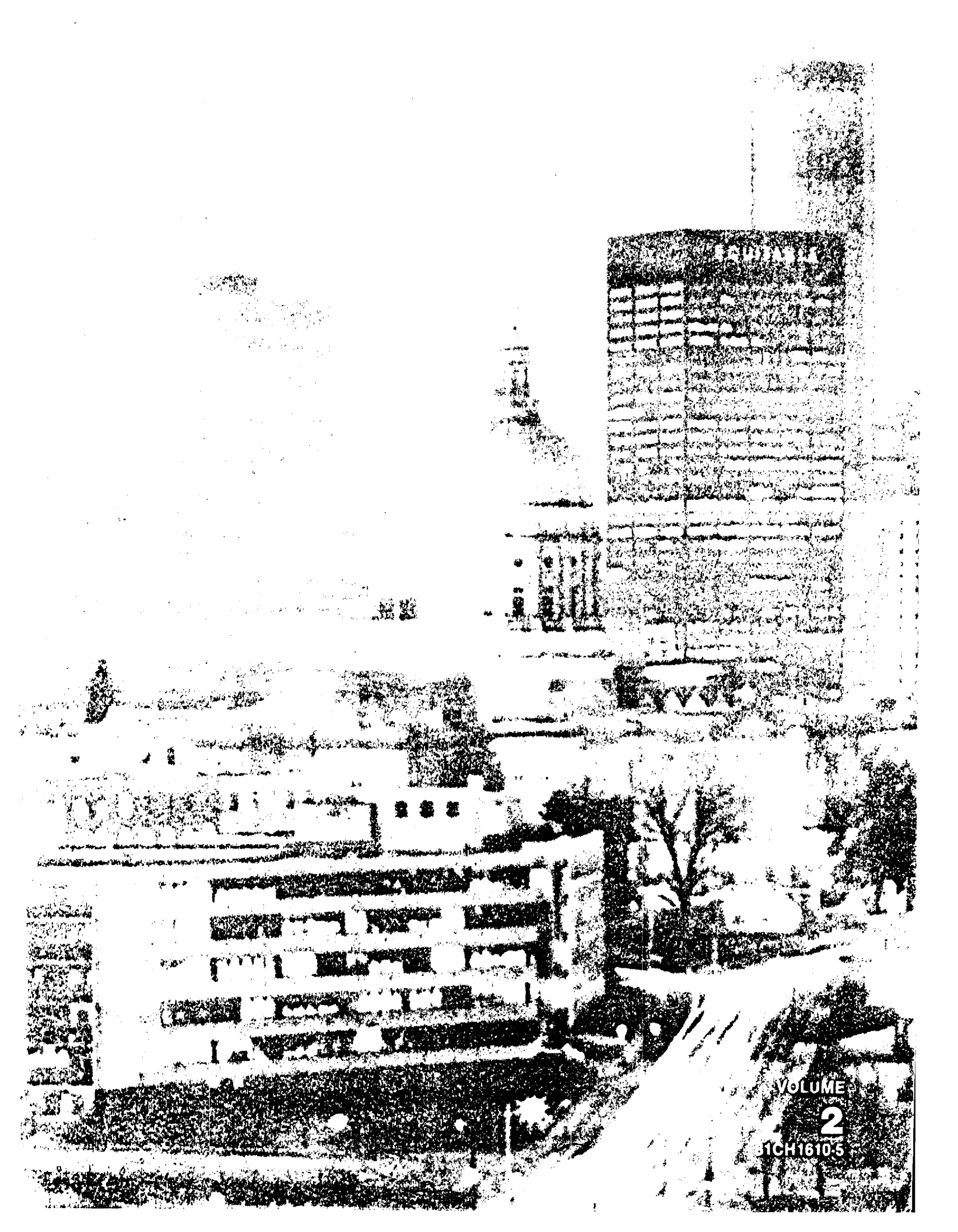


Figure 1. Speech Production Model.  
(After Portnoff (4))



VOLUME

2

31CH1610-5

EVALUATION OF TWO-DIMENSIONAL DISCRETE FOURIER  
TRANSFORMS VIA GENERALIZED FFT ALGORITHMS

Theresa C. Speake and Russell M. Mersereau

School of Electrical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332

ABSTRACT

In this paper two-dimensional fast Fourier transforms (FFT's) are expressed as special cases of a generalization of the one-dimensional Cooley-Tukey algorithm. This generalized algorithm allows the efficient evaluation of discrete Fourier transforms (DFT's) of rectangularly sampled sequences, hexagonally sampled sequences and arbitrary periodically sampled sequences. Significant computational savings can be realized using this generalized algorithm when the periodicity matrix of the sequence is highly composite. Alternate factorizations of the periodicity matrix lead to different FFT algorithms, including the row-column decomposition and the vector-radix algorithm. This paper will present a generalized DFT, derive the general 2-D Cooley-Tukey algorithm and conclude by interpreting several 2-D FFT algorithms in terms of the generalized one.

INTRODUCTION

An important problem in digital signal processing is the efficient calculation of two-dimensional discrete Fourier transforms (DFT's). These transforms are generally computed either as a series of 1-D DFT's obtained from a row-column decomposition of the DFT sum or as a series of smaller 2-D DFT's via the vector-radix algorithm (1). However, both of these approaches are special cases of a generalization of the Cooley-Tukey algorithm (2) to the 2-D case. This generalization is the subject of this paper. Other special cases of this algorithm can be used to evaluate DFT's of hexagonally sampled sequences (3) or other periodically sampled sequences.

The key to the efficiency of the 1-D FFT is the significant computational savings realized when  $N$ , the length of the transform, is a highly composite number. The 2-D counterpart of  $N$  is an integer matrix  $\underline{N}$ , called the periodicity matrix, that depends upon the support of the sequence. Alternate factorizations of this matrix lead to different FFT algorithms. As in the 1-D case, the efficiency of the calculation is related to the

compositeness of  $\underline{N}$ . Two specific factorizations result in the row-column decomposition and the vector-radix algorithm. If sampling strategies other than rectangular are used, different periodicity matrices are necessary but the resulting DFT summations have the same structure. Thus, the generalized Cooley-Tukey algorithm can be used for these calculations as well.

A GENERAL DISCRETE FOURIER TRANSFORM

A periodic 2-D sequence is one which repeats itself at regularly spaced intervals in each of two independent directions. A 2-D sequence  $\tilde{x}(n_1, n_2)$  is periodic if it has the property

$$\tilde{x}(\underline{n} + \underline{N} \underline{r}) = \tilde{x}(\underline{n}) \quad (1)$$

for all integer vectors  $\underline{n}$  and  $\underline{r}$  and some  $2 \times 2$  integer matrix  $\underline{N}$  with a nonzero determinant. A 2-D periodic sequence repeats itself in the directions defined by the vectors forming the columns of the periodicity matrix  $\underline{N}$ . If  $\underline{N}$  is diagonal, the sequence it describes is said to be rectangularly periodic. For all periodicity matrices the determinant of  $\underline{N}$ ,  $\det \underline{N}$ , is an integer whose absolute value is equal to the number of samples in one period of  $\tilde{x}(\underline{n})$ .

Any periodic sequence  $\tilde{x}(\underline{n})$  with periodicity matrix  $\underline{N}$  can be exactly represented by a set of Fourier series coefficients,  $\tilde{X}(\underline{k})$ . Thus,

$$\tilde{x}(\underline{n}) = \frac{1}{|\det \underline{N}|} \sum_{\underline{k} \in \underline{J}_N} \tilde{X}(\underline{k}) \exp(j \underline{k}' (2\pi \underline{N}^{-1}) \underline{n}) \quad (2)$$

$$\tilde{X}(\underline{k}) = \sum_{\underline{n} \in \underline{I}_N} \tilde{x}(\underline{n}) \exp(-j \underline{k}' (2\pi \underline{N}^{-1}) \underline{n}) \quad (3)$$

The sequence of coefficients  $\tilde{X}(\underline{k})$  is periodic with periodicity matrix  $\underline{N}'$  where ' denotes matrix transposition. The regions  $\underline{I}_N$  and  $\underline{J}_N$  denote the set of samples in one period of  $\tilde{x}(\underline{n})$  and  $\tilde{X}(\underline{k})$ , respectively.

We can define a generalized DFT by recognizing the one-to-one correspondence between finite sequences with support in  $\underline{I}_N$  and periodic sequences with periodicity matrix  $\underline{N}$  that have one period of

\*This work was supported, in part, by the National Science Foundation under grant ECS-7817201 and by the Joint Services Electronics Program Contract DAAG29-78-C-0005.

samples contained in  $I_{\underline{N}}$ . Specifically, if  $x(\underline{n})$  denotes a sequence with finite support on  $I_{\underline{N}}$ , then

$$\tilde{x}(\underline{n}) = \sum_{\underline{r}} x(\underline{n} + \underline{N} \underline{r}) \quad (4)$$

where  $\underline{r}$  varies over all 2-D integer vectors and

$$x(\underline{n}) = \begin{cases} \tilde{x}(\underline{n}), & \underline{n} \in I_{\underline{N}} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

Thus,  $x(\underline{n})$  is completely specified by  $\tilde{x}(\underline{n})$ ,  $\tilde{x}(\underline{n})$  is completely specified by  $X(\underline{k})$  and  $X(\underline{k})$  is completely specified by  $X(\underline{k})$ , a finite extent sequence with support on  $J_{\underline{N}}$ . The DFT concisely states this relationship:

$$X(\underline{k}) = \sum_{\underline{n} \in I_{\underline{N}}} x(\underline{n}) \exp(-j \underline{k}' (2\pi \underline{N}^{-1}) \underline{n}), \quad \underline{k} \in J_{\underline{N}} \quad (6)$$

$$x(\underline{n}) = \frac{1}{|\det \underline{N}|} \sum_{\underline{k} \in J_{\underline{N}}} X(\underline{k}) \exp(j \underline{k}' (2\pi \underline{N}^{-1}) \underline{n}), \quad \underline{n} \in I_{\underline{N}} \quad (7)$$

The remainder of this paper will be concerned with the efficient evaluation of Eq. (6). The algorithm to be presented here will permit the calculation of Eq. (6) irrespective of the periodicity matrix  $\underline{N}$ .

#### THE GENERAL COOLEY-TUKEY ALGORITHM

Cooley-Tukey type FFT algorithms exist for the evaluation of Eq. (6) whenever  $\underline{N}$  can be factored into a non-trivial product of integer matrices. This is consistent with the existence condition for a 1-D FFT which requires that the length of the transform be a composite integer. As in the 1-D case, the more factors in  $\underline{N}$ , the greater the computational savings. Since  $\underline{N}$  is an integer matrix, it contains non-trivial integer matrix factors whenever  $\det \underline{N}$  is a non-prime number (4). Thus,  $\underline{N}$  can be factored into a product of two integer matrices

$$\underline{N} = \underline{P} \underline{Q} \quad (8)$$

It should be noted that the factorization in Eq. (8) is not unique and that it is ordered.

We shall say that two integer vectors  $\underline{m}$  and  $\underline{n}$  are congruent to one another with respect to a matrix modulus  $\underline{N}$  if

$$\underline{m} = \underline{n} + \underline{N} \underline{r} \quad (9)$$

for some integer vector  $\underline{r}$ . The indices of the samples of a periodic sequence are thus congruent, with respect to the periodicity matrix, to the equivalent samples on other periods of the sequence. Specifically, every sample of  $x(\underline{n})$  is con-

gruent to a sample from  $I_{\underline{N}}$ . We shall use the notation

$$\underline{m} = ((\underline{n}))_{\underline{N}} \quad (10)$$

to denote that vector which is both congruent to  $\underline{n}$  and contained in  $I_{\underline{N}}$ .

If  $\underline{N}$  satisfies Eq. (8), any vector  $\underline{n}$  in  $I_{\underline{N}}$  can be uniquely expressed as

$$\underline{n} = ((\underline{P} \underline{q} + \underline{p}))_{\underline{N}} \quad (11)$$

where  $\underline{p} \in I_{\underline{P}}$  and  $\underline{q} \in I_{\underline{Q}}$ .  $I_{\underline{P}}$  and  $I_{\underline{Q}}$  are sets containing  $|\det \underline{P}|$  and  $|\det \underline{Q}|$  members, respectively. More will be said later about  $I_{\underline{P}}$  and  $I_{\underline{Q}}$ . Any pair of vectors, one from  $I_{\underline{P}}$  and one from  $I_{\underline{Q}}$ , defines a unique vector  $\underline{n}$  from  $I_{\underline{N}}$  according to Eq. (11). Similarly,  $\underline{k}'$  can be expressed as

$$\underline{k}' = ((\underline{l}' + \underline{m}' \underline{Q}))_{\underline{N}'} \quad (12)$$

where  $\underline{m} \in J_{\underline{P}}$  and  $\underline{l}' \in J_{\underline{Q}}$ .  $J_{\underline{P}}$  and  $J_{\underline{Q}}$  are two sets of vectors in the frequency domain whose elements, when combined according to Eq. (12), form all of the elements in  $J_{\underline{N}}$ .  $J_{\underline{P}}$  and  $J_{\underline{Q}}$  contain  $|\det \underline{P}|$  and  $|\det \underline{Q}|$  members, respectively.

Using Eqs. (11) and (12), the DFT sum in Eq. (6) can be written as

$$X(\underline{Q}' \underline{m} + \underline{l}') = \sum_{\underline{p}, \underline{q}} x((\underline{P} \underline{q} + \underline{p}))_{\underline{N}} \exp[-j(\underline{l}' + \underline{m}' \underline{Q})' (2\pi \underline{N}^{-1}) (\underline{P} \underline{q} + \underline{p})] \quad (13)$$

By expanding the exponential term, this sum can be written in two parts,

$$C(\underline{p}, \underline{l}) = \sum_{\underline{q} \in I_{\underline{Q}}} x((\underline{P} \underline{q} + \underline{p}))_{\underline{N}} \exp[-j(\underline{l}' + \underline{m}' \underline{Q})' (2\pi \underline{Q}^{-1}) \underline{q}] \quad (14a)$$

$$X(\underline{Q}' \underline{m} + \underline{l}') = \sum_{\underline{p} \in I_{\underline{P}}} C(\underline{p}, \underline{l}) \exp[-j \underline{l}' (2\pi \underline{N}^{-1}) \underline{p}] \exp[-j \underline{m}' (2\pi \underline{P}^{-1}) \underline{p}] \quad (14b)$$

These relations represent the first level of decomposition of a decimation-in-time Cooley-Tukey FFT algorithm. If  $\underline{P} \underline{Q} = \underline{Q} \underline{P}$ , a similar algorithm corresponding to decimation-in-frequency can be derived.

To understand Eqs. (14), consider the two separately. The sequence  $x((\underline{P} \underline{q} + \underline{p}))_{\underline{N}}$ , interpreted as a sequence over the variable  $\underline{q}$ , is periodic with periodicity matrix  $\underline{Q}$ . Thus, the summation in (14a) represents a 2-D DFT of the array  $x((\underline{P} \underline{q} + \underline{p}))_{\underline{N}}$

taken with respect to the periodicity matrix  $\underline{Q}$ . The region of support for this sequence is  $I_{\underline{Q}}$  which must be chosen as one period of  $x((\underline{P}\underline{q} + \underline{p}))_{\underline{N}}$ , interpreted as a function of  $\underline{q}$ . A different matrix- $\underline{Q}$  DFT must be evaluated for each value of the vector  $\underline{p}$ . Thus,  $|\det \underline{P}|$  such transforms need to be evaluated.

The summation in Eq. (14b) shows how the outputs of these matrix- $\underline{Q}$  DFT's should be combined to form the matrix- $\underline{N}$  DFT. The numbers  $C(\underline{p}, \underline{\ell})$  are multiplied by the factors  $\exp[-j\underline{\ell}'(2\pi\underline{N}^{-1})\underline{p}]$ , called twiddle factors, and the products are combined in a series of matrix- $\underline{P}$  DFT's or butterflies. There are  $|\det \underline{N}|$  number of twiddle factor multiplications and  $|\det \underline{Q}|$  number of matrix- $\underline{P}$  butterflies. If either  $\underline{P}$  or  $\underline{Q}$  is factorable, either set of smaller DFT's can be further decomposed.

The vectors

$$\underline{n} = ((\underline{P}\underline{q}))_{\underline{N}} \quad (15)$$

form that subset of  $I_{\underline{N}}$  which is created by sampling  $I_{\underline{N}}$  with the sampling matrix  $\underline{P}$ . For a fixed value of  $\underline{p}$ , the samples

$$\underline{n} = ((\underline{P}\underline{q} + \underline{p}))_{\underline{N}}, \quad \underline{q} \in I_{\underline{Q}} \quad (16)$$

form a coset with respect to this subset. Since each coset is the same size as the subset, there are  $|\det \underline{N}|/|\det \underline{Q}| = |\det \underline{P}|$  cosets in all. The members of any coset are congruent to one another with respect to the modulus  $\underline{P}$ . The region  $I_{\underline{P}}$  should be chosen to consist of one member from each coset or  $|\det \underline{P}|$  elements in all.

The regions  $J_{\underline{P}}$  and  $J_{\underline{Q}}$  are chosen similarly. Since the Fourier transform is periodic in  $\underline{k}$  with periodicity matrix  $\underline{N}'$  and

$$\begin{aligned} \underline{N}' &= \underline{Q}'\underline{P} \\ \underline{k} &= \underline{Q}'\underline{m} + \underline{\ell}, \end{aligned} \quad (17)$$

$\underline{Q}'$  in the frequency domain plays an analogous role to  $\underline{P}$  in the spatial domain and  $\underline{P}'$  plays an analogous role to  $\underline{Q}$ .

#### AN EXAMPLE

To illustrate the general FFT algorithm, consider the evaluation of a 4x4 rectangular DFT with periodicity matrix  $\underline{N}$  using the factorization  $\underline{P}\underline{Q}$ :

$$\underline{N} = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix} \underline{P} = \begin{bmatrix} 2 & 2 \\ 1 & -1 \end{bmatrix} \underline{Q} = \begin{bmatrix} 1 & 2 \\ 1 & -2 \end{bmatrix}$$

Figure 1 shows the region  $I_{\underline{N}}$  divided into four cosets by the sampling matrix  $\underline{P}$  with a different symbol indicating the members of each coset. Notice that all four cosets have the same geometry when periodically extended. To define  $I_{\underline{Q}}$  we need a set of four vectors which satisfy Eq. (15) by span

ning any one of the cosets. One possible set of vectors is

$$I_{\underline{Q}} = [(0,0)', (1,0)', (2,0)', (3,0)']$$

The set  $I_{\underline{P}}$  must consist of one member of each coset. One possibility is to set  $I_{\underline{P}} = I_{\underline{Q}}$ .

Figure 2 shows the region of support for the DFT,  $J_{\underline{N}}$ , divided into four cosets by the frequency domain sampling matrix  $\underline{Q}'$ . From this figure we see that possible choices for  $J_{\underline{P}}$  and  $J_{\underline{Q}}$  are

$$\begin{aligned} J_{\underline{P}} &= [(0,0)', (1,0)', (2,0)', (3,0)'] \\ J_{\underline{Q}} &= [(0,0)', (0,1)', (0,2)', (0,3)'] \end{aligned}$$

Once these four sets are chosen, the algorithm is determined. Each matrix- $\underline{Q}$  DFT operates on one of the cosets of the input, shown in Figure 1, to produce an intermediate array. That array is multiplied by the twiddle factors (all equal to one for this DFT) and the results become the inputs to the matrix- $\underline{P}$  DFT's. Each of the latter DFT's produces one of the output cosets shown in Figure 2.

The four outputs of the matrix- $\underline{Q}$  DFT's can be computed directly from the four inputs. If the inputs are denoted by  $w, x, y,$  and  $z$  and the outputs by  $A, B, C,$  and  $D$ , the direct evaluation of these DFT's corresponds to setting

$$\begin{aligned} A &= w + x + y + z \\ B &= w - jx - y + jz \\ C &= w - x + y - z \\ D &= w + jx - y - jz \end{aligned} \quad (18)$$

Alternatively, since  $\underline{Q}$  is composite, we could use the factorization

$$\underline{Q} = \begin{bmatrix} 1 & 2 \\ 1 & -2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}. \quad (19)$$

The flow chart of a four-point DFT based on this factorization is shown in Figure 3.

For this particular example the matrix- $\underline{P}$  DFT's are very similar and can be evaluated using Eq. (18). The inputs and outputs may be arranged in such a way that the flow chart in Figure 3 describes both the matrix- $\underline{Q}$  DFT's and the matrix- $\underline{P}$  DFT's.

#### RELATIONSHIP TO STANDARD 2-D FFT'S

The two common FFT algorithms for evaluating the DFT's of rectangularly sampled data are the row-column decomposition (1) and the vector-radix algorithm (2). Both of these can be described in terms of the generalized Cooley-Tukey algorithm. The periodicity matrix of a 2-D rectangular DFT has the form

$$\underline{N} = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix}$$

with the sets  $I_{\underline{N}}$  and  $J_{\underline{N}}$  chosen to represent rectangularly shaped regions of support:

$$I_{\underline{N}} = J_{\underline{N}} = [(m_1, m_2)'] : 0 \leq m_1 < N_1$$

and  $0 \leq m_2 < N_2$

The basic row-column algorithm corresponds to the factorization

$$\underline{N} = \underline{P} \underline{Q} = \begin{bmatrix} N_1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & N_2 \end{bmatrix}$$

With this factorization the column transforms are calculated first, followed by the row transforms. We can identify the sets

$$I_{\underline{P}} = J_{\underline{P}} = [(p_1, 0)'] : 0 \leq p_1 < N_1$$

$$I_{\underline{Q}} = J_{\underline{Q}} = [(0, q_2)'] : 0 \leq q_2 < N_2$$

If  $N_1$  and  $N_2$  are each divisible by two, we have the alternate factorization

$$\underline{P} \underline{Q} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} N_1/2 & 0 \\ 0 & N_2/2 \end{bmatrix}$$

In this case we have

$$I_{\underline{P}} = J_{\underline{P}} = [(0,0)', (0,1)', (1,0)', (1,1)']$$

$$I_{\underline{Q}} = J_{\underline{Q}} = [(q_1, q_2)'] : 0 \leq q_1 < N_1/2$$

and  $0 \leq q_2 < N_2/2$

This factorization of  $\underline{N}$  corresponds to the first stage of decimation for a vector-radix algorithm.

#### SUMMARY

The most useful feature of this generalized Cooley-Tukey algorithm is that all of the information concerning the efficiency of an algorithm is contained in the periodicity matrix. This allows the comparison and optimization of several algorithms by examination of the factorization of  $\underline{N}$ .

#### REFERENCES

- (1) D. B. Harris, J. H. McClellan, D. S. K. Chan and H. W. Schuessler, "Vector radix fast Fourier transform," 1977 IEEE Int. Conf. on ASSP Record, pp. 548-551, 1977.
- (2) J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier

series," Math. Comput., vol. 19, pp. 296-301, 1965.

- (3) R. M. Mersereau, "The processing of hexagonally sampled two-dimensional signals," Proc. IEEE, vol. 67, pp. 930-949, 1979.
- (4) R. M. Mersereau and Theresa C. Speake, "Cooley-Tukey algorithms for the evaluation of multi-dimensional discrete Fourier transforms," submitted to IEEE Trans., ASSP, 1980.

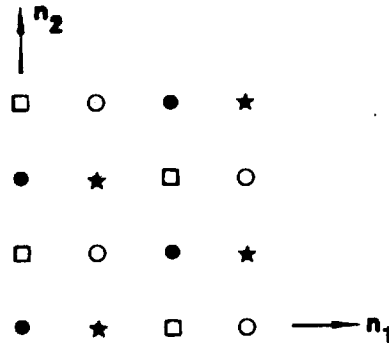


Figure 1. The spatial domain region of support,  $I_{\underline{N}}$ , for a 4X4 FFT divided into four cosets of samples.

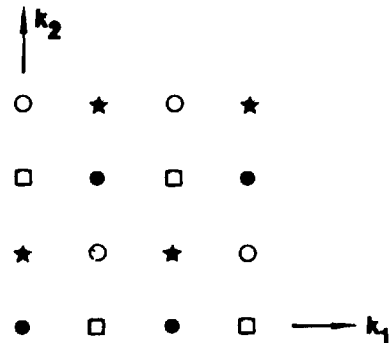


Figure 2. The frequency domain region of support,  $J_{\underline{N}}$ , for a 4X4 FFT divided into four cosets of samples.

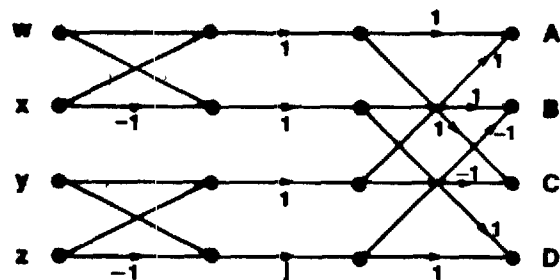
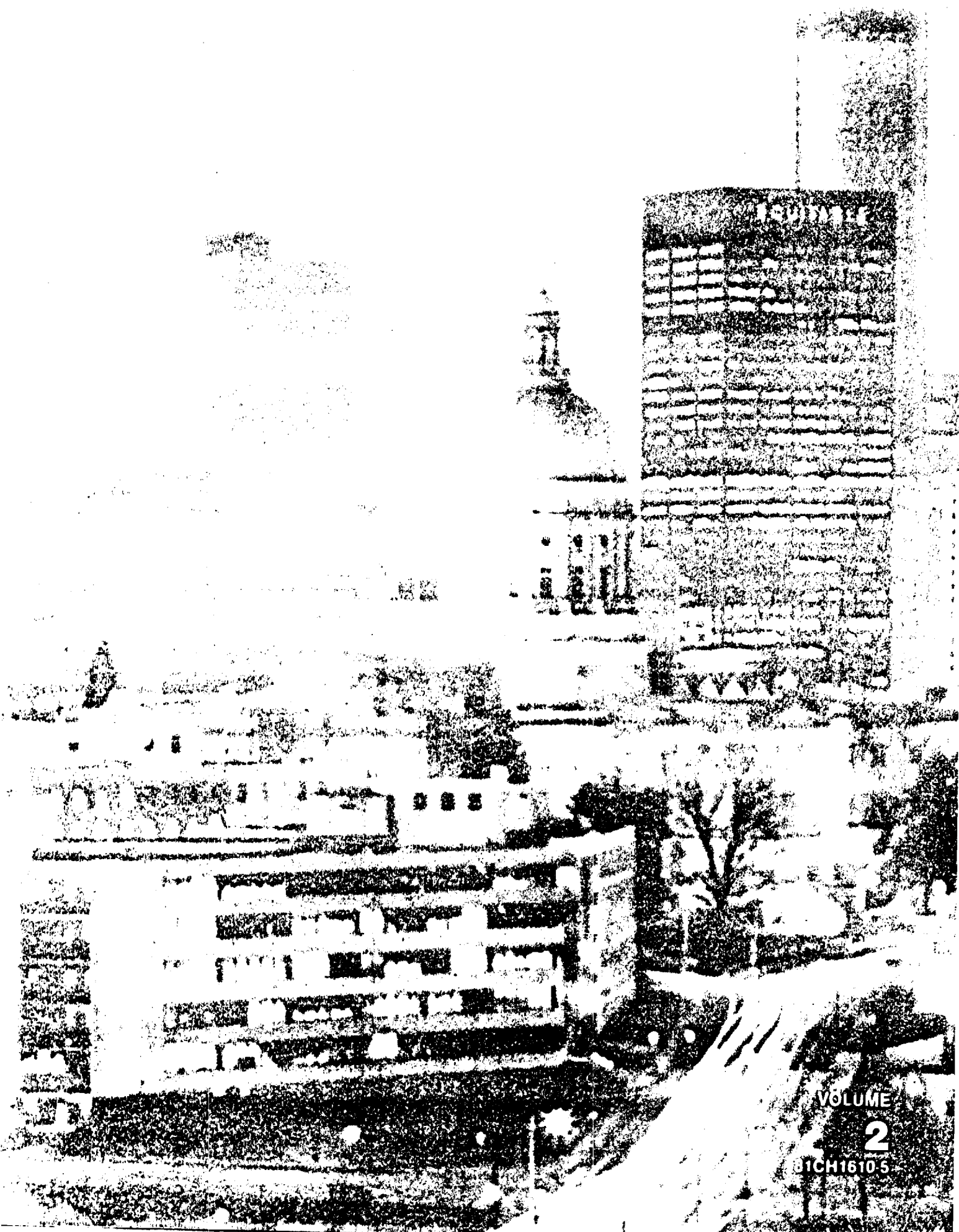


Figure 3. A flow chart for the evaluation of a matrix-Q DFT using the decomposition in (19).



BRUNNEN

VOLUME

2

31CH16105

AN INTERPOLATION TECHNIQUE FOR PERIODICALLY  
SAMPLED TWO-DIMENSIONAL SIGNALS\*

Theresa C. Speake and Russell M. Mersereau

School of Electrical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332

ABSTRACT

Two-dimensional signals are usually represented for the purpose of digital processing as rectangularly sampled functions of two orthogonal independent variables. It has been previously noted that sampling a signal on a hexagonal grid can offer substantial savings in digital storage and computation. In this paper these two sampling schemes will be generalized, via a matrix description, to include arbitrarily sampled 2-D signals. Using this matrix description and a generalized discrete Fourier transform, a technique will be presented for interpolating a set of sample points from one sampling grid to an alternative one. Since the analog signal from which the original samples were obtained is frequently unavailable for resampling, the ability to easily convert from one sampling scheme to another can be important in the efficient processing of a particular signal.

INTRODUCTION

The representation of bandlimited signals as arrays of numbers is fundamental to digital signal processing. In making the generalization from one-dimensional to two-dimensional signals a number of possibilities exist for the representation, or sampling, of the analog signal. The most common sampling scheme for two-dimensional signals is the rectangular or row-column one. In this scheme the signal is sampled at evenly spaced values of each of two orthogonal independent variables. While this is the most common generalization of one-dimensional periodic sampling, it is neither the only nor, necessarily, the most efficient one. Petersen and Middleton(1) showed that rectangular sampling is a special case of a more general sampling of signals on a non-orthogonal set of axes. Mersereau (2) considered the problems associated with processing signals sampled on a hexagonal grid. In this paper we will extend these ideas to a general matrix description of two-dimensional sampling.

\*This work was supported, in part, by the National Science Foundation under grant ECS-7817201 and by the Joint Services Electronics Program contract DAAG29-78-C-0005.

Using this matrix description and the generalized discrete Fourier transform (DFT) (3), we will present an interpolation technique for obtaining an alternate representation of a bandlimited signal given only the samples from a particular periodic grid. The interpolation referred to here is a change in the location of the samples while maintaining the sample density. If a change in the sample density is desired, interpolation techniques such as those given by Schafer and Rabiner (4) can be used.

To define the sampled signal, we begin with the analog signal  $x_a(t)$  where  $t=[t_1, t_2]'$  represents the independent variables defining  $x_a$  and ' denotes matrix transposition. The operation of periodic sampling can be described by

$$x(\underline{n}) = x_a(\underline{V}\underline{n}) \quad (1)$$

where  $\underline{n} = [n_1, n_2]'$  is an integer vector and  $\underline{V}$  is the  $2 \times 2$  sampling matrix. The columns of  $\underline{V}$ ,  $\underline{v}_1$  and  $\underline{v}_2$ , must be linearly independent and thus the determinant of  $\underline{V}$ ,  $\det \underline{V}$ , will be nonzero. The magnitudes of  $\underline{v}_1$  and  $\underline{v}_2$  represent the sampling periods in the  $\underline{v}_1$  and  $\underline{v}_2$  directions, respectively.

It is well-known that the process of sampling an analog signal has the effect of replicating the Fourier transform of the signal in the frequency domain. This replication can be expressed in terms of a matrix  $\underline{U}$ , called the aliasing matrix, whose columns indicate the replication directions and periods. Given  $X(\underline{\omega})$  is the Fourier transform of  $x(\underline{n})$ ,

$$X(\underline{\omega}) = \sum_{\underline{n}} x(\underline{n}) \exp(-j\underline{\omega}'\underline{n}), \quad (2)$$

and  $X_a(\underline{\Omega})$  is the Fourier transform of  $x_a(t)$ , then the two transforms are related by

$$X(\underline{\omega}) = X(\underline{V}'\underline{\Omega}) = \frac{1}{|\det \underline{V}|} \sum_{\underline{k}} X_a(\underline{\Omega} - \underline{U}\underline{k}) \quad (3)$$

where  $\underline{k}$  is an integer vector.  $\underline{U}$ , the aliasing matrix, is related to  $\underline{V}$ , the sampling matrix, by

$$\underline{U}'\underline{V} = 2\pi\underline{I} \quad (4)$$

where  $\underline{I}$  is the identity matrix. The Fourier transform,  $X(\underline{\omega})$ , of  $x(\underline{n})$  can be interpreted as a periodic extension of  $X_a(\underline{\Omega})$  with the periodicity described by the column vectors,  $\underline{u}_1$  and  $\underline{u}_2$ , of  $\underline{U}$ .

In order to recover  $x(t)$  from its samples,  $x(n)$ , it is necessary to restrict  $x(t)$  to be bandlimited in the frequency domain. That is

$$X_a(\Omega) = 0, \text{ for } \Omega \notin B$$

for some finite region  $B$  in the frequency domain. For  $x(t)$  to be recoverable from  $x(n)$ , the aliasing matrix  $\underline{U}$  must be such that the replicated sequences in Eq. (3) do not overlap. This implies a restriction on the sampling matrix  $\underline{V}$  consistent with Eq. (4). Although  $\underline{U}$  describes the replication of  $X_a(\Omega)$ , the requirement of no overlapping, or aliasing, does not lead to a unique choice of  $\underline{U}$ . However, by also imposing the restriction that  $x_a(t)$  be represented by as few samples per unit area as possible, we obtain the additional requirement that  $|\det \underline{U}|$  be minimized. Thus, for an efficient and invertible sampling scheme for a bandlimited signal, the aliasing matrix  $\underline{U}$  should be chosen to have the smallest possible value of  $|\det \underline{U}|$  that avoids aliasing for the particular region  $B$ .

Petersen and Middleton (1) showed that a bandlimited signal with a circular band region can be represented by 13.4% fewer hexagonally arranged samples than rectangular ones. This savings can result in a significant increase in computational efficiency for a number of signal processing applications (5).

#### SAMPLING IN THE FOURIER DOMAIN

To take into account the particular sampling grid used to obtain  $x(n)$  from the analog signal, we will define the Fourier transform of  $x(n)$  as

$$X(\underline{V}'\Omega) = \sum_{\underline{n}} x(n) \exp(-j\underline{\Omega}'\underline{V}\underline{n}) = X_{\underline{V}}(\underline{\Omega}) \quad (5)$$

and the inverse transform as

$$x(n) = \frac{|\det \underline{V}|}{4\pi^2} \int_B X(\underline{V}'\Omega) \exp(j\underline{\Omega}'\underline{V}\underline{n}) d\underline{\Omega} \quad (6)$$

where  $B$  is the region in the frequency domain representing one of the periods defined by  $\underline{U}$ . The Fourier transform, a continuous function of  $\underline{\Omega}$ , is periodic with periodicity defined by  $\underline{U}$ . Thus,

$$X_{\underline{V}}(\underline{\Omega} + \underline{U}\underline{n}) = X_{\underline{V}}(\underline{\Omega}) \quad (7)$$

As in (3), let us consider a general periodic discrete sequence  $\tilde{x}(n)$  with periodicity matrix  $\underline{N}$ . That is,

$$\tilde{x}(n) = \tilde{x}(n + \underline{N}\underline{r}) \quad (8)$$

where  $\underline{r}$  is any integer vector and  $\underline{N}$  is a  $2 \times 2$  integer matrix with a nonzero determinant whose columns,  $\underline{n}_1$  and  $\underline{n}_2$ , define the directions and periods of repetition of  $\tilde{x}(n)$ . Alternately,  $\tilde{x}(n)$  can be thought of as a finite sequence extended according to  $\underline{N}$  to form a periodic sequence. Given the discrete Fourier transform (DFT) defined in (3),

$$X(k) = \sum_{\underline{n}} x(n) \exp(-jk'(2\pi\mathbf{N}^{-1})\underline{n}), \quad k \in J_{\underline{N}} \quad (9)$$

$$x(n) = \frac{1}{|\det \underline{N}|} \sum_{\underline{k}} X(k) \exp(jk'(2\pi\mathbf{N}^{-1})\underline{n}), \quad n \in I_{\underline{N}} \quad (10)$$

each value of  $\underline{N}$  represents a different DFT for the original finite sequence  $x(n)$ . The difference in these DFT's is in the locations of the Fourier transform samples defined in Eq. (9). One interpretation of Eq. (9) is that it represents the samples of the Fourier transform of a continuous, bandlimited signal that has been periodically sampled. The matrix  $\underline{N}$  specifies the locations of these Fourier transform samples. Comparing the Fourier transform of  $x(n)$  given in Eq. (5) and the DFT in Eq. (9) yields,

$$X(k) = X(\underline{\omega}) \Big|_{\underline{\omega} = (2\pi\mathbf{N}^{-1})'k} \quad (11)$$

Defining  $\underline{R} = (2\pi\mathbf{N}^{-1})'$  to be the Fourier domain sampling matrix leads to a Fourier domain analogy with the sampling matrix  $\underline{V}$ . Using the interpretation of the DFT as a sampled Fourier transform, the vectors defined by the columns of  $\underline{R}$  define the locations of the transform samples in the  $\underline{\omega}$ -plane.

We can now recognize a duality between sampling in the spatial domain and in the frequency domain. In the spatial domain, the sampling matrix  $\underline{V}$  specifies the sample locations of the analog signal. The aliasing matrix  $\underline{U}$  specifies the repetition of the Fourier transform of the original signal in forming the Fourier transform of the sampled signal.  $\underline{U}$  and  $\underline{V}$  are related by

$$\underline{U}'\underline{V} = 2\pi\mathbf{I}.$$

The DFT coefficients given in (9) can be interpreted as samples of the Fourier transform of the sequence  $x(n)$  or as samples of the periodically extended Fourier transform of a continuous, bandlimited signal. The matrix specifying the locations of the transform samples,  $\underline{R}$ , is thus analogous to  $\underline{V}$  while the matrix defining the periodic extension or repetition of  $x(n)$ ,  $\underline{N}$ , is analogous to  $\underline{U}$ . By definition,  $\underline{N}$  and  $\underline{R}$  are related by

$$\underline{N}'\underline{R} = 2\pi\mathbf{I}$$

In determining an interpolation scheme for converting from one periodic representation to another, it will be useful to have an alternate definition of the frequency sampling matrix,  $\underline{R}$ . Referring to Eq. (11), we can make substitution  $\underline{\omega} = \underline{V}'\underline{\Omega}$  and evaluate the Fourier transform for values of the analog frequency,  $\underline{\Omega}$ . Thus,

$$X(k) = X(\underline{V}'\underline{\Omega}) \Big|_{\underline{V}'\underline{\Omega} = (2\pi\mathbf{N}^{-1})'k}$$

where

$$\begin{aligned} \underline{\Omega} &= (\underline{V}')^{-1} (2\pi \underline{N}^{-1})' \underline{k} \\ &= \underline{U} (\underline{N}^{-1})' \underline{k} \\ &= \underline{R} \underline{k} \end{aligned} \quad (13)$$

This definition of the Fourier domain sampling matrix is a function of the periodicity of the signal in both the spatial domain and the Fourier domain.

#### GENERAL INTERPOLATION SCHEME

Assume  $x_{V_1}(\underline{n})$  is discrete array, sampled according to  $\underline{V}_1^{-1}$ , that represents a bandlimited analog signal,  $x_a(\underline{t})$ . Most digital signal processing systems assume a discrete array as input. However, an alternative discrete representation of  $x_a(\underline{t})$  may result in a more computationally efficient processing system. Since the original analog signal is seldom available for resampling, we would like to be able to obtain an alternate discrete representation for  $x_a(\underline{t})$  by discrete array  $x_{V_2}(\underline{n})$  given only  $x_{V_1}(\underline{n})$ .

In most cases interpolation would be used to reduce the sample density in the discrete representation of the original signal. In this paper we have divided the interpolation problem into two parts. One part is the simple reduction of sampling rate by an integer factor which has been discussed in the 1-D case (4). This part of the interpolation can be generalized from the 1-D case. The other part can be referred to as geometric interpolation and is concerned with obtaining samples on an alternative grid while maintaining the sample density. It is this part of the interpolation problem that is of interest here.

The basic approach to interpolation depends on the defining equation for the Fourier domain sampling matrix,

$$\underline{R} = \underline{U} (\underline{N}^{-1})'$$

If we can find integer matrices  $\underline{N}_1$  and  $\underline{N}_2$  such that

$$\underline{U}_1 (\underline{N}_1^{-1})' = \underline{U}_2 (\underline{N}_2^{-1})', \quad (14)$$

then  $x_a(\underline{t})$  sampled according to  $\underline{U}_1$  and periodically extended according to  $\underline{N}_1$  has its DFT samples in the same locations in the  $\underline{u}$ -plane as if it had been sampled according to  $\underline{U}_2$  and extended according to  $\underline{N}_2$ . Given  $\underline{U}_1$  and  $\underline{U}_2$  and the restriction that  $\underline{N}_1$  and  $\underline{N}_2$  be integer valued, the solution of Eq. (14) leads to an interpolation algorithm. Namely, compute the DFT of  $x_{V_1}(\underline{n})$  using the periodicity matrix  $\underline{N}_1$ , then calculate the inverse transform according to the periodicity matrix  $\underline{N}_2$ . The result of the inverse transform is the sample sequence  $x_{V_2}(\underline{n})$ . In terms of the DFT definition,

$$X_1(\underline{k}) = \sum_{\underline{n}} x_{V_1}(\underline{n}) \exp(-j2\pi \underline{k}' \underline{N}_1^{-1} \underline{n}), \quad \underline{n} \in I_{\underline{N}_1} \quad (15)$$

is the DFT of  $x_{V_1}(\underline{n})$  extended according to  $\underline{N}_1$ . To obtain the alternatively sampled sequence,

$$x_{V_2}(\underline{n}) = \frac{1}{|\det \underline{N}_2|} \sum_{\underline{k}} X_1(\underline{k}) \exp(j2\pi \underline{k}' \underline{N}_2^{-1} \underline{n}), \quad \underline{k} \in J_{\underline{N}_2} \quad (16)$$

is the inverse DFT of  $X_1(\underline{k})$ . Depending upon the compositeness of  $\underline{N}_1$  and  $\underline{N}_2$ , either or both of these calculations can be performed with an FFT algorithm.

#### AN EXAMPLE

As an illustration of the interpolation scheme, consider the interpolation of a rectangularly sampled sequence to the corresponding hexagonally sampled sequence. We will assume the original analog signal,  $x_a(\underline{t})$ , is circularly bandlimited with cutoff frequency of  $|\underline{\Omega}| = \pi$  radians. Sampling the signal with the most efficient and invertible rectangular scheme requires the sampling matrix,

$$\underline{V}_R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

and an aliasing matrix,

$$\underline{U}_R = \begin{bmatrix} 2\pi & 0 \\ 0 & 2\pi \end{bmatrix}.$$

We would like to represent  $x_a(\underline{t})$  in terms of hexagonally arranged samples. A sampling matrix that results in a hexagonal sampling grid is

$$\underline{V}_H = \begin{bmatrix} 1 & -\frac{1}{2} \\ 0 & 1 \end{bmatrix}$$

with a corresponding aliasing matrix of

$$\underline{U}_H = \begin{bmatrix} 2\pi & \pi \\ 0 & 2\pi \end{bmatrix}$$

This hexagonal sampling scheme has the same sampling density as the rectangular scheme.

If the rectangular  $M \times M$ -point sequence is extended rectangularly,

$$\underline{N}_R = \begin{bmatrix} M & 0 \\ 0 & M \end{bmatrix},$$

an integer matrix solution exists for Eq. (14) when  $M$  is an even integer. The resultant hexagonal periodicity matrix is

$$\underline{N}_H = \begin{bmatrix} M & 0 \\ M/2 & M \end{bmatrix}$$

A DFT of the original rectangularly sampled sequence, calculated using  $N_H$ , yields samples of the Fourier transform at locations in the  $\Omega$ -plane specified by the Fourier domain sampling matrix

$$\underline{k} = \begin{bmatrix} 2\pi/M & 0 \\ 0 & 2\pi/M \end{bmatrix}$$

Then by calculating the inverse DFT using  $N_H$ , a set of hexagonal samples is obtained to represent the original analog signal.

For the particular example considered here, several simplifications are possible to reduce the required DFT's to one-dimensional calculations. Substituting Eq. (16) into Eq. (15) yields the interpolation formula,

$$x_{\underline{V}_H}(n) = \sum_{\underline{m}} x_{\underline{V}_R}(m) \sum_{\underline{k}} \frac{1}{|\det N_H|} \exp[j2\pi \underline{k}'(N_H^{-1}n - N_R^{-1}m)] \quad (17)$$

For the particular periodicity matrices  $N_R$  and  $N_H$ , the second summation in (17), the interpolation function, is equal to

$$\frac{1}{M^2} \sum_{k_1} \exp[j2\pi(k_1((2n_1 - n_2 - 2m_1)/2M) + k_2((n_2 - m_2)/M))] \quad (18)$$

The function in (18) can be simplified by noting that the second term is nonzero for  $n_2 = m_2$  only. Thus, the interpolation function reduces to

$$\frac{1}{N} \sum_{k_1} \exp[j2\pi k_1((2n_1 - n_2 - 2m_1)/2M)] \quad (19)$$

and the complete interpolation equation is

$$x_{\underline{V}_H}(n) = \sum_{m_1} x_{\underline{V}_R}(m_1, n_2) \frac{1}{N} \sum_{k_1} \exp[j2\pi k_1((2n_1 - n_2 - 2m_1)/2M)] \quad (20)$$

By interchanging the order of summation, Eq. (20) can be calculated with two one-dimensional FFT's for each row of  $x_{\underline{V}_H}(n)$ . That is,

$$x_{\underline{V}_H}(n) = \frac{1}{N} \sum_{k_1=0}^{2M-1} \exp[j2\pi k_1((2n_1 - n_2)/2M)] \sum_{m_1=0}^{M-1} x_{\underline{V}_R}(m_1, n_2) \exp[-j2\pi k_1 m_1/M] \quad (21)$$

The right-hand summation can be calculated as an  $M$ -point 1-D FFT on the rows of  $x_{\underline{V}_R}$  to form a function  $f(k_1, n_2)$ . Then by augmenting  $f(k_1, n_2)$  with zeros to form a  $2M$ -point sequence, the left-hand

summation can be calculated as a  $2M$ -point inverse 1-D FFT on the rows of  $f(k_1, n_2)$ . The desired hexagonal sequence consists of every other point of the inverted sequence. For this particular example, the rectangular samples and the hexagonal samples are the same for the even rows of the original sequence. Thus, the calculation in Eq.(21) must be performed for the odd rows of  $x_{\underline{V}_R}(n)$  only.

#### SUMMARY

In this paper we have outlined a method for converting a set of points representing an analog signal sampled on one particular grid to the corresponding samples taken on an alternative grid. The basic approach makes use of a matrix description of sampling in the spatial and Fourier domains. A simple example illustrated the method by obtaining a hexagonal representation of a circularly bandlimited signal from its rectangular samples.

Several problems remain for further study. In Eq. (14) conditions for the existence of an integer matrix solution for  $N_1$  and  $N_2$  have not been developed. In addition, for interpolation examples more complicated than the one considered here, the regions of support ( $I_{N_1}$  in Eq. (15) and  $J_{N_2}$  in Eq. (16)) of the forward and inverse DFT's become more difficult to define. Finally, the attractiveness of this interpolation technique lies in its calculation via an FFT algorithm. Since the efficiency of the FFT is dependent upon the compositeness of  $N_1$  and  $N_2$ , the efficiency of the interpolation scheme varies with the particular sampling grids considered.

#### REFERENCES

- (1) D.P. Petersen and D. Middleton, "Sampling and Reconstruction of Wave-Number-Limited Functions in N-Dimensional Euclidean Spaces", Information and Control, vol. 5, pp.279-323, 1962.
- (2) R.M. Mersereau, "The Processing of Hexagonally Sampled Two-Dimensional Signals", Proc. of the IEEE, vol. 67, pp. 930-949, June 1979.
- (3) Theresa C. Speake and Russell M. Mersereau, "Evaluation of Two-Dimensional Discrete Fourier Transforms via Generalized FFT Algorithms", Proc. 1981 IEEE Int. Conf. on Acoustics, Speech and Signal Processing, March 1981.
- (4) Ronald W. Schafer and Lawrence R. Rabiner, "A Digital Signal Processing Approach to Interpolation", Proc. of the IEEE, Vol. 61, pp. 692-702, June 1973.
- (5) Russell M. Mersereau, Tae H. Joo and Theresa C. Speake, "A Comparison of Hexagonally and Rectangularly Sampled Two-Dimensional FIR Digital Filters", Proc. 1980 IEEE Int. Conf. on Acoustics, Speech and Signal Processing, April 1980.



INTERNATIONAL CONFERENCE  
ON  
**DIGITAL SIGNAL PROCESSING**

**Sponsored by:**

Imperial College of Science and Technology, University of London,  
England

Facoltà di Ingegneria, Università di Firenze, Florence, Italy  
Istituto di Ricerca sulle Onde Elettromagnetiche del C.N.R., Firenze

EURASIP - European Association for Signal Processing

Sezione di Firenze dell'A.E.I.

Sezione di Firenze dell'A.N.I.P.L.A.

A. I. C. A.

A. I. T. A.

I.E.E.E. - Middle & South Italy Section

I.E.E.E. - North Italy Section

In cooperation with the I.E.E.E. Audio Speech and Signal  
Processing Society

President: B. McA. Sayers, Imperial College, London

Conference Co-Chairmen: V. Cappellini, University of Florence  
A. G. Constantinides, Imperial College,  
London

## **PROCEEDINGS**

**PALAZZO DEI CONGRESSI**

Florence, Italy



September 2-5, 1981

**MULTI-DIMENSIONAL DIGITAL SIGNAL PROCESSING FROM  
ARBITRARY PERIODIC SAMPLING RASTERS**

**Russell M. Mersereau  
Theresa C. Speake**

**School of Electrical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332 USA**

**This work was supported in part by National Science Foundation Grant  
ECS-7817201 and Joint Services Electronics Program Contract DAA29-78-  
C-0005.**

Multi-dimensional signals such as photographic images and electromagnetic field distributions can be sampled on periodic, but non-rectangular, sampling lattices. These representations are important for both sampling and processing such signals because they often require a lower sampling density than the comparable rectangular lattice. This means less storage and fewer computations. The most common of these alternative sampling strategies uses hexagonal sampling to represent two-dimensional isotropically bandlimited signals.

This talk is concerned with non-rectangular sampling and with algorithms for processing the resulting arrays. It will begin with the presentation of a matrix notation for discussing both rectangular and non-rectangular sampling. With this notation, multi-dimensional extensions of a number of one-dimensional algorithms and formalisms can be readily derived. Then linear filtering of non-rectangularly sampled signals will be discussed. The final part of the talk will discuss discrete spectral analysis. Included in this portion of the talk will be a discussion of non-rectangular discrete Fourier transforms and non-rectangular Cooley-Tukey (FFT) algorithms.

#### I. NON-RECTANGULAR SAMPLING

If  $x_a(t_1, t_2)$  denotes a spatially continuous two-dimensional signal, the operation of rectangular sampling can be described by

$$x(n_1, n_2) = x_a(n_1 T_1, n_2 T_2), \quad (1)$$

where  $T_1$  and  $T_2$  are the horizontal and vertical sampling intervals. If  $x_a(t_1, t_2)$  is bandlimited such that its Fourier transform  $X_a(\omega_1, \omega_2)$  satisfies  $X_a(\omega_1, \omega_2) = 0$  for  $|\omega_1| \geq \frac{\pi}{T_1}$ ,  $|\omega_2| \geq \frac{\pi}{T_2}$ , then  $x_a(t_1, t_2)$  can be exactly recovered from its rectangular samples. This result is well known. Furthermore, if the Fourier transform of the sequence  $x(n_1, n_2)$  is defined as

$$X(\omega_1, \omega_2) = \sum_{n_1} \sum_{n_2} x(n_1, n_2) \exp[-jn_1\omega_1 T_1 - jn_2\omega_2 T_2] \quad (2)$$

then the Fourier transforms of the sequence and that of the spatially continuous waveform will be related by

$$X(\omega_1, \omega_2) = \frac{1}{T_1 T_2} \sum_{r_1} \sum_{r_2} X_a\left(\omega_1 - \frac{2\pi r_1}{T_1}, \omega_2 - \frac{2\pi r_2}{T_2}\right) \quad (3)$$

The sequence Fourier transform is periodic in a rectangular sense with period  $\frac{2\pi}{T_1}$  in  $\omega_1$  and  $\frac{2\pi}{T_2}$  in  $\omega_2$ .

Notationally, these expressions can be simplified by adopting a vector notation for the signals. Thus, by defining  $\underline{t} = (t_1, t_2)'$ ,  $\underline{\omega} = (\omega_1, \omega_2)'$  etc. (' denotes transposition) eqs. (1)-(3) become

$$x(\underline{n}) = x_a(\underline{T} \underline{n}) \quad (4)$$

$$X(\underline{\omega}) = \sum_{\underline{n}} x(\underline{n}) \exp[-j\underline{\omega}' \underline{T} \underline{n}] \quad (5)$$

$$X(\underline{\omega}) = \frac{1}{|\det \underline{T}|} \sum_{\underline{r}} X_a(\underline{\omega} - (2\pi \underline{T}'^{-1}) \underline{r}) \quad (6)$$

The matrix  $\underline{T}$  is known as the sampling matrix. For the rectangular sampling lattice defined in (1), it is given by

$$\underline{T} = \begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix} \quad (7)$$

Eqs. (4)-(6) can also be used to describe M-dimensional rectangular sampling. In this case  $\underline{T}$  becomes an M x M diagonal matrix and  $\underline{n}$ ,  $\underline{u}$ , and  $\underline{r}$  become M element column vectors.

Periodic, non-rectangular sampling can also be described by Eqs. (4)-(6). The only difference is that  $\underline{T}$  is no longer diagonal. In fact, the columns of  $\underline{T}$  are vectors whose integer linear combinations define the sampling lattice. The Fourier transform  $X(\underline{\omega})$  is periodic in  $\underline{\omega}$  with a periodicity matrix  $\underline{U} = 2\pi\underline{T}^{-1}$ . By this we mean

$$X(\underline{\omega}) = X(\underline{\omega} + \underline{U} \underline{k})$$

for any integer vector  $\underline{k}$ . If  $x_a(\underline{t})$  is bandlimited such that its Fourier transform is confined to one period of  $X(\underline{\omega})$ , then  $x_a(\underline{t})$  can be recovered exactly from  $x(\underline{n})$ . It is interesting to note that the sampling density is given by  $|\det \underline{U}| = \frac{1}{|\det \underline{T}|}$  which is equal to the area of one period of  $X(\underline{\omega})$ .

Hexagonal sampling corresponds to the sampling matrix

$$\underline{T} = \begin{bmatrix} \frac{T_1}{2} & \frac{T_1}{2} \\ T_2 & -T_2 \end{bmatrix}$$

It is optimal for representing bandlimited, spatially continuous signals whose Fourier transforms are confined to an ellipse. Of all sampling lattices which permit an exact reconstruction of the signal, the hexagonal one has the minimum sampling density.

## II. FILTERING NON-RECTANGULARLY SAMPLED SIGNALS

Let us assume that we have a system which accepts a sequence  $x(\underline{n})$  (sampling matrix  $\underline{T}$ ) as its input and produces a sequence  $y(\underline{n})$  as its output. (Sampling matrix  $\underline{T}$  also). If that system is linear and shift-invariant then the output is the input convolved with the impulse response of the system,  $h(\underline{n})$ , (defined with respect to the sampling matrix  $\underline{T}$ ). Thus

$$y(\underline{n}) = \sum_{\underline{k}} x(\underline{k}) h(\underline{n}-\underline{k}) \quad (8)$$

The form of the convolution is completely independent of  $\underline{T}$ .

A sampled complex sinusoid of the form  $x(\underline{n}) = \exp [j\underline{\omega}'\underline{T} \underline{n}]$  is an eigenvector of the sequence. That is, when the input is a complex sinusoid, the output will be one also. The eigenvalue or gain at that frequency is the frequency response.

$$H(\underline{\omega}) = \sum_{\underline{n}} h(\underline{n}) \exp [-j\underline{\omega}'\underline{T} \underline{n}] \quad (9)$$

which is the Fourier transform of the impulse response, just as in the rectangular and one-dimensional cases. Eq. (9) can be inverted using

$$h(\underline{n}) = \frac{1}{|\det \underline{U}|} \int_{I_{\underline{U}}} H(\underline{\omega}) \exp [j\underline{\omega}'\underline{V}\underline{n}] d\underline{\omega} \quad (10)$$

where

$$\underline{T}'\underline{U} = 2\pi\underline{I}$$

and  $\underline{I}$  is the identity matrix.  $I_{\underline{U}}$  is one period of the periodic function  $H(\underline{\omega})$ . (periodicity matrix  $\underline{U}$ ).

The Fourier transform defined in (5) satisfies all of the properties associated with its rectangular counterpart. Among the more important of these are the fact that it is a linear transform and that when sequences are convolved according to (8), their Fourier transforms multiply. A version of Parseval's relation can also be defined.

### III. NON-RECTANGULAR DISCRETE FOURIER TRANSFORMS

A sequence,  $\tilde{x}(\underline{n})$ , is periodic if it satisfies the relation

$$\tilde{x}(\underline{n} + \underline{M} \underline{r}) = \tilde{x}(\underline{n}) \quad (11)$$

$$\det \underline{M} \neq 0$$

for all integer vectors  $\underline{n}$  and  $\underline{r}$ . The integer matrix  $\underline{M}$  is called the periodicity matrix. The number of samples in one period of  $\tilde{x}(\underline{n})$  is given by  $|\det \underline{M}|$ . For a given periodic sequence the periodicity matrix is not unique.

The periodic sequence  $\tilde{x}(\underline{n})$  can be uniquely represented as a finite sum of harmonically related complex exponentials

$$\tilde{x}(\underline{n}) = \frac{1}{|\det \underline{M}|} \sum_{\underline{k} \in J_{\underline{M}}} \tilde{X}(\underline{k}) \exp [j\underline{k}'(2\pi\underline{M}^{-1})\underline{n}] \quad (12)$$

where

$$\tilde{X}(\underline{k}) = \sum_{\underline{n} \in I_{\underline{M}}} \tilde{x}(\underline{n}) \exp [-j\underline{k}'(2\pi\underline{M}^{-1})\underline{n}] \quad (13)$$

$J_{\underline{M}}$  is a set of  $|\det \underline{M}|$  members in the  $\underline{k}$ -domain.  $I_{\underline{M}}$  represents the set of samples in one period of  $\tilde{x}(\underline{n})$ .

There is a one-to-one correspondence between sequences with finite support which are confined to  $I_{\underline{M}}$  and periodic sequences with periodicity matrix  $\underline{M}$ . Because of this relationship it follows that  $x(\underline{n})$  can be exactly represented by the coefficients  $\tilde{X}(\underline{k})$  of the Fourier series. This relationship becomes the discrete Fourier transform (DFT)

$$X(\underline{k}) = \sum_{\substack{\underline{n} \in I_{\underline{M}}}} x(\underline{n}) \exp[-j\underline{k}'(2\pi\underline{M}^{-1})\underline{n}] , \quad \underline{k} \in J_{\underline{M}} \quad (14)$$

$$x(\underline{n}) = \frac{1}{|\det \underline{M}|} \sum_{\substack{\underline{k} \in J_{\underline{M}}}} X(\underline{k}) \exp[j\underline{k}'(2\pi\underline{M}^{-1})\underline{n}] , \quad \underline{n} \in I_{\underline{M}} . \quad (15)$$

The numbers  $X(\underline{k})$  can be interpreted as samples of the Fourier transform of  $x(\underline{n})$ . In particular

$$X(\underline{k}) = \hat{X}(\underline{T}^{-1})' (2\pi\underline{M}^{-1})' \underline{k} , \quad (16)$$

where  $\hat{X}(\omega)$  is the Fourier transform of  $x(\underline{n})$ . Thus the locations of the DFT samples in the Fourier space depend both upon the sampling matrix  $\underline{T}$  and the periodicity matrix  $\underline{M}$ .

Some special cases of (14) are sufficiently important to justify elucidation. Rectangular sampling, as we have already seen corresponds to the special case where  $\underline{T}$  is diagonal. If  $\underline{M}$  is also diagonal

$$\underline{M} = \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} \quad (17)$$

we have the normal 2-D DFT which corresponds to a rectangularly sampled

Fourier transform. If  $\underline{N}$  is of the form

$$\underline{N} = \begin{bmatrix} N_1 & N_1 \\ N_2 & -N_2 \end{bmatrix} \quad (18)$$

we get hexagonal samples of the Fourier transform of a rectangularly sampled sequence. A periodicity matrix of the form

$$\underline{N} = \begin{bmatrix} N_1 & N_2 \\ N_2 & N_1 \end{bmatrix} \quad (19)$$

corresponds to the hexagonally sampled Fourier transform of a hexagonally sampled sequence.

We know that the 1-D DFT can be evaluated efficiently using the Cooley-Tukey algorithm whenever the length of the transform,  $N$ , is a composite integer. The greater the number of factors that can be found for  $N$ , the greater the computational savings. Cooley-Tukey type algorithms for multi-dimensional DFT's can be found whenever the integer periodicity matrix  $\underline{N}$  is a highly composite matrix. A matrix is composite if it can be factored into two other integer matrices

$$\underline{N} = \underline{P} \underline{Q} \quad (20)$$

such that  $|\det \underline{P}| < |\det \underline{N}|$   
 $|\det \underline{Q}| < |\det \underline{N}|$ .

The details of this algorithm will not be presented here, but they are available in the literature.

Traditional algorithms for evaluating multi-dimensional DFTs can

be interpreted in terms of factoring the periodicity matrix. The row-column algorithm for evaluating the rectangular DFT corresponds to the factorization

$$\underline{M} = \underline{P} \underline{Q} = \begin{bmatrix} N_1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (21)$$

and the vector radix algorithm corresponds to the factorization

$$\begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} \frac{N_1}{2} & 0 \\ 0 & \frac{N_2}{2} \end{bmatrix} \quad (22)$$

A number of questions in the general area of non-rectangular sampling are worthy of further study. Although a number of them are concerned with the details of one or another of the algorithms which have already been derived, a very fundamental question remains to be answered -- how are these sampling strategies and the resulting algorithmic complexity ultimately related? We have evidence that some sampling procedures are better than others in certain situations, but the issue of optimizing the sampling strategy for anything other than minimum sampling density has not been addressed. We have a means for producing an unlimited number of FFT algorithms to evaluate any DFT, for example. These algorithms differ in their computational complexity. Yet without an exhaustive enumeration, how can we tell which is best? This is typical of the tantalizing questions which remain.

EVALUATION OF MULTIDIMENSIONAL DFT'S ON ARBITRARY SAMPLING LATTICES\*

R.M. Mersereau, E.W. Brown, III and A. Guessoum

School of Electrical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332

The discrete Fourier transform (DFT) can be generalized to the multidimensional case in a natural way which allows for discrete transform representations of signals which are defined on arbitrary periodic sampling lattices. In this paper it is shown that these generalized DFTs can be evaluated using algorithms for evaluating one-dimensional DFT's, such as Winograd's Fourier transform algorithm. This approach would seem to have some computational advantages over the lattice Cooley-Tukey algorithm presented by Mersereau and Speake.<sup>1</sup>

Introduction

This paper addresses the problem of evaluating a general multidimensional discrete Fourier transform (DFT) of the form

$$X(k) = \sum_{n \in I_N} x(n) \exp[-jk^T(2\pi N^{-1}n)]. \quad (1)$$

The sequence  $x(n)$  and its DFT,  $X(k)$ , are assumed to be  $M$ -dimensional. Thus,  $n$  and  $k$ , the signal domain and Fourier domain independent variables are  $M$ -dimensional column vectors with integer coefficients. The non-zero samples of  $x(n)$  are confined to the region  $I_N$  in the signal domain. The matrix  $N$  is known as the periodicity matrix. It is an  $M \times M$  matrix with integer elements whose role in the multi-dimensional DFT is analogous to the transform length of a one-dimensional algorithm. For a traditional multidimensional DFT,  $N$  is diagonal, but non-diagonal periodicity matrices can occur in computing the DFT of a signal which is not rectangularly sampled. For example, a two-dimensional DFT which relates a rectangularly sampled signal to rectangular samples of its Fourier transform uses the periodicity matrix

$$\underline{N} = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (2)$$

\*This work was supported, in part, by the National Science Foundation under grant ECS-7817201 and by the Joint Services Electronics Program under contract DAAG29-81-K-0024.

and one which relates a hexagonally sampled signal to hexagonal samples of its Fourier transform<sup>2</sup> uses the periodicity matrix

$$\underline{N} = \begin{bmatrix} 2N & N \\ N & 2N \end{bmatrix} \quad (3)$$

A detailed derivation of (1) is presented in a paper which should be available early in 1983<sup>3</sup>.

MATRIX COOLEY-TUKEY ALGORITHM

In this section we will outline a matrix generalization of the Cooley-Tukey fast Fourier transform (FFT) algorithm,<sup>4</sup> which can be used to evaluate (1). A more complete discussion is given in Reference [1].

The key to the efficiency of the generalized Cooley-Tukey algorithm is the factorability of the periodicity matrix,  $\underline{N}$ . It is well known that the efficiency of a 1-D FFT algorithm depends strongly upon the length of the transform,  $N$ . These algorithms become truly efficient only when  $N$  is a highly composite integer. Similarly, efficient Cooley-Tukey algorithms for the multidimensional problem exist whenever the periodicity matrix,  $\underline{N}$ , is a composite integer matrix. If  $\underline{N}$  is composite, it can be written as

$$\underline{N} = \underline{P}\underline{Q} \quad (4)$$

where  $\underline{P}$  and  $\underline{Q}$  are integer matrices such that  $|\det \underline{P}| \geq 2$  and  $|\det \underline{Q}| \geq 2$ . (As an aside it can be noted that  $\underline{N}$  is factorable whenever the absolute value of its determinant, which must be an integer, is not a prime number. Such a factorization is not unique, except possibly in the one-dimensional case).

The summation in (1) produces a distinct value for  $|\det \underline{N}|$  different values of  $k$  and it is invertible if the region  $I_N$  contains the same number of samples. The  $|\det \underline{N}|$  values of  $n$  and  $k$  can be expressed as

$$k = \underline{Q}m + \underline{1} + \underline{N}r \quad (5a)$$

$$n = \underline{P}q + \underline{p} + \underline{N}s \quad (5b)$$

where  $\underline{p}$  and  $\underline{m}$  come from sets of integer vectors containing  $|\det \underline{P}|$  members and  $\underline{q}$  and  $\underline{1}$  come from

sets of integer vectors containing  $|\det \underline{Q}|$  members. Substituting equations (5) into (1) reduces the matrix- $\underline{N}$  DFT into  $|\det \underline{P}|$  matrix- $\underline{Q}$  DFTs,  $|\det \underline{Q}|$  matrix- $\underline{P}$  DFTs, and  $|\det \underline{N}|$  complex multiplications. The net computational effort is less than if (1) is evaluated directly. If either  $\underline{P}$  or  $\underline{Q}$  is composite a similar decomposition can be used in evaluating the smaller DFTs.

Two of the more common algorithms for evaluating the multidimensional rectangular DFT, the row-column algorithm and the vector-radix algorithm<sup>5</sup>, correspond to special cases of this algorithm. The row-column algorithm corresponds to the factorization (in the 2-D case)

$$\underline{N} = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} = \begin{bmatrix} N_1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (6)$$

and the radix  $(2 \times 2)$  vector-radix algorithm corresponds to the factorization

$$\begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \dots \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \quad (7)$$

While the generalized FFT algorithm is elegant from a conceptual point of view, it is difficult to implement for non-diagonal periodicity matrices. The difficulty is with the vector equivalent of the operation of bit-reversal. A resolution of this difficulty is known for the hexagonal case but the resulting algorithm requires that data be reindexed at each decimation stage in the algorithm. The techniques described in the next section reduce the amount of data shuffling required.

#### SMITH'S NORMAL FORM

When  $\underline{N}$  is diagonal, a multidimensional DFT can be efficiently implemented using either the row-column algorithm or the vector-radix algorithm. If  $\underline{N}$  is non-diagonal, we can develop similar algorithms if we first write it in Smith's normal form

$$\underline{N} = \underline{U} \underline{D} \underline{V} \quad (8)$$

where  $\underline{D}$  is an integer diagonal matrix and  $\underline{U}$  and  $\underline{V}$  are unimodular, i.e.,  $|\det \underline{U}| = |\det \underline{V}| = 1$  and  $\underline{U}$  and  $\underline{V}$  are integer matrices. This decomposition can be performed by executing elementary row and column operations on  $\underline{N}$ .

Substituting eq. (8) into eq. (1), the DFT summation can be written as

$$\underline{X}(\underline{k}) = \sum_{\underline{n} \in I_{\underline{N}}} \underline{x}(\underline{n}) \exp[-j \underline{k}^T \underline{V}^{-1} (2\pi \underline{D}^{-1}) \underline{U}^{-1} \underline{n}]. \quad (9)$$

Define

$$\underline{\hat{n}} = \underline{U}^{-1} \underline{n} \quad (10a)$$

$$\underline{\hat{k}} = \underline{V}^{-T} \underline{k} \quad (10b)$$

Then the DFT summation reduces to

$$\underline{\hat{X}}(\underline{\hat{k}}) = \sum_{\underline{n} \in I_{\underline{N}}} \underline{x}(\underline{\hat{n}}) \exp[-j \underline{\hat{k}}^T (2\pi \underline{D}^{-1}) \underline{\hat{n}}] \quad (11)$$

This sum is seen to be a matrix- $\underline{D}$  DFT. Furthermore since the inverse of a unimodular matrix is unimodular,  $\underline{n}$  and  $\underline{\hat{n}}$  (and similarly  $\underline{k}$  and  $\underline{\hat{k}}$ ) describe the same sampling lattice. The sequence  $\underline{x}(\underline{\hat{n}})$  is simply a reindexing of the samples of  $\underline{x}(\underline{n})$ . This decomposition provides the following algorithm for evaluating a matrix- $\underline{N}$  DFT:

1. Express  $\underline{N}$  in Smith's normal form as  $\underline{N} = \underline{U} \underline{D} \underline{V}$ .
2. Scramble the input sequence according to the relation  $\underline{\hat{n}} = \underline{U}^{-1} \underline{n}$ .
3. Compute a DFT of the resulting sequence using the diagonal periodicity matrix  $\underline{D}$ . Call the result  $\underline{\hat{X}}(\underline{\hat{k}})$ .
4. Unscramble the output sequence according to the relation  $\underline{k} = \underline{V}^T \underline{\hat{k}}$ .

Observe that with this algorithm the multidimensional arrays need to be reindexed at most twice, once at the beginning of the algorithm and once at the end.

While the matrix  $\underline{D}$  is an  $M \times M$  integer matrix, the matrix- $\underline{D}$  DFT at step 3 of the algorithm is not necessarily an  $M$ -dimensional DFT. To illustrate this fact consider a two-dimensional matrix- $\underline{N}$  DFT for which  $|\det \underline{N}| = N_1 N_2$ . For some  $\underline{N}$  the matrix  $\underline{D}$  will assume the form

$$\underline{D} = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} \quad (12)$$

We will say that such a  $\underline{D}$  matrix is of Form A. A  $\underline{D}$ -matrix of Form A corresponds to an  $(N_1 \times N_2)$ -point rectangular two-dimensional DFT. Alternatively  $\underline{D}$  could be of the following form, which we call Form B:

$$\underline{D} = \begin{bmatrix} N_1 N_2 & 0 \\ 0 & 1 \end{bmatrix} \quad (13)$$

Such a  $\underline{D}$  matrix corresponds to an  $N_1 N_2$ -point one-dimensional DFT. The algorithm given in this section will work with either form for  $\underline{D}$  but when  $\underline{D}$  is of Form A, the resulting algorithm is more efficient.

The Smith normal form representation for an integer matrix is not unique but generally an  $N$ -matrix of Form A cannot be converted into one of Form B and vice versa. One exception to this rule occurs when  $N_1$  and  $N_2$  are relatively prime, a fact which is exploited by Good's prime factor algorithm<sup>6</sup>. In a similar fashion, when  $N$  is  $M$ -dimensional, the dimensionality of the matrix- $D$  DFT may vary from 1 to  $M$ .

#### DISCUSSION

In addition to its value as a computational tool, the existence of the  $UDV$  algorithm for evaluating a matrix- $N$  DFT establishes some bounds on the efficiency of any algorithm for evaluating (1). If algorithmic complexity is measured by the number of required multiplies, we see that steps (2) and (4) of the  $UDV$  algorithm are multiply-free. The number of multiplies required to evaluate a matrix- $N$  DFT is thus equal to the number required to evaluate a matrix- $D$  DFT. That complexity, in turn, is intermediate between the complexity of a 1-D algorithm with  $|\det N|$  points and an  $M$ -dimensional algorithm with  $|\det N|$  points. Furthermore the complexity of a rectangular (i.e. diagonal  $N$ ) DFT algorithm can be no better than that of an algorithm for a non-diagonal  $N$ . This must be true since we can write

$$\underline{D} = \underline{U}^{-1} \underline{N} \underline{V}^{-1} \quad (14)$$

Thus, results, such as those by Winograd<sup>7</sup> and Nussbaumer<sup>8</sup>, on the complexity of 1-D DFT algorithms and rectangular DFT algorithms can also be applied to algorithms involving non-diagonal periodicity matrices.

An open question with the matrix Cooley-Tukey algorithm concerns choosing the matrix factors  $\underline{P}$  and  $\underline{Q}$ , since, in general, the factorization of an integer matrix is not unique. From the above results we see that the efficiency of the resulting algorithm depends not on the specific choices for  $\underline{P}$  and  $\underline{Q}$  but rather on whether their Smith normal forms are of Form A or Form B.

A number of open questions remain. Among these are whether  $\underline{U}$  and  $\underline{V}$  can be chosen to minimize the amount of data shuffling and whether that shuffling can be combined with the data manipulation required by the matrix- $D$  DFT's.

#### REFERENCES

- [1] R.M. Mersereau and T.C. Speake, "A unified treatment of Cooley-Tukey algorithms for the evaluation of the multidimensional DFT," IEEE Trans. Acoustics, Speech, Signal Processing, vol. ASSP-29, No. 5, pp. 1011-1018, Oct. 1981.
- [2] R.M. Mersereau, "The processing of hexagonally sampled two-dimensional signals," Proc. IEEE, vol. 67, pp. 930-949, June 1979.

- [3] R.M. Mersereau and T.C. Speake, "The processing of periodically sampled multidimensional signals," IEEE Trans. Acoustics, Speech, Signal Processing, vol. ASSP-31, (to appear).
- [4] J.W. Cooley and J.W. Tukey, "An algorithm for the machine calculation of complex Fourier series," Math. Comput., vol. 19, no. 90, pp. 296-301, 1965.
- [5] D.B. Harris, J.H. McClellan, D.S.K. Chan, and H.W. Schuessler, "Vector radix fast Fourier transform," in 1977 IEEE Int. Conf. Acoust., Speech, Signal Processing Rec., 1977, pp. 548-551.
- [6] I.J. Good, "The interaction algorithm and practical Fourier series," J. Royal Stat. Soc., ser. B, 20 (1958), pp. 361-72; Addendum, 22 (1960), pp. 372-75.
- [7] S. Winograd, "On computing the discrete Fourier transform," Proc. Nat. Acad. Sci., USA, 73, pp. 1005-1006, 1976.
- [8] H.J. Nussbaumer and P. Quandalle, "Fast computation of discrete Fourier transforms using polynomial transforms," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-27, pp. 169-181, 1979.

ROW-COLUMN ALGORITHMS FOR THE EVALUATION OF MULTIDIMENSIONAL  
DFT'S ON ARBITRARY PERIODIC SAMPLING LATTICES\*

R. M. Mersereau, E. W. Brown, III, and A. Guessoum

School of Electrical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332

ABSTRACT

Recent work by Mersereau and Speake [1,2] has shown that multidimensional discrete Fourier transforms (DFTs) can be defined for signals defined on any periodic sampling lattice and that they can be evaluated using a generalization of the Cooley-Tukey FFT algorithm. The main purpose of this work was to develop alternative algorithms which were more suitable to highly parallel machine architectures and which required less data handling than the Cooley-Tukey algorithms. Such an algorithm is described here. It makes use of the Smith normal form representation of an integer matrix. As a sidelight to this work a Chinese remainder theorem for lattices has been developed which permits an extension of Good's prime factor algorithm. This is also described.

INTRODUCTION

This paper addresses the problem of evaluating a general multidimensional discrete Fourier transform (DFT) of the form

$$X(\mathbf{k}) = \sum_{\mathbf{n} \in I_{\mathbf{N}}} x(\mathbf{n}) \exp[-j\mathbf{k}^T (2\pi \mathbf{N}^{-1}) \mathbf{n}]. \quad (1)$$

The sequence  $x(\mathbf{n})$  and its DFT,  $X(\mathbf{k})$ , are assumed to be  $M$ -dimensional. Thus,  $\mathbf{n}$  and  $\mathbf{k}$ , the signal domain and Fourier domain independent variables are  $M$ -dimensional column vectors with integer coefficients. The non-zero samples of  $x(\mathbf{n})$  are confined to the region  $I_{\mathbf{N}}$  in the signal domain. The matrix  $\mathbf{N}$  is known as the periodicity matrix. It is an  $M \times M$  matrix with integer elements whose role in the multi-dimensional DFT is analogous to the transform length of a one-dimensional algorithm. For the traditional DFT, which relates a rectangularly sampled signal to rectangular samples of its Fourier transform,  $\mathbf{N}$  is diagonal, but non-diagonal periodicity matrices can occur in computing the DFT of a signal which is not rectangularly sampled. For example, a two-dimensional DFT which relates a hexagonally sampled signal to hexagonal samples of its Fourier transform uses the periodicity matrix

$$\mathbf{N} = \begin{bmatrix} 2N & N \\ N & 2N \end{bmatrix} \quad (2)$$

Such a DFT is derived and discussed in [3].

MATRIX COOLEY-TUKEY ALGORITHM

In this section we will outline a matrix generalization of the Cooley-Tukey fast Fourier transform (FFT) algorithm [4] which can be used to evaluate (1). A more complete discussion is given in [2].

The key to the efficiency of the generalized Cooley-Tukey algorithm is the factorability of the periodicity matrix,  $\mathbf{N}$ . It is well known that the efficiency of a 1-D FFT algorithm depends strongly upon the length of the transform,  $N$ ; these algorithms become truly efficient only when  $N$  is a highly composite integer. Similarly, efficient Cooley-Tukey algorithms for the multi-dimensional problem exist whenever the periodicity matrix,  $\mathbf{N}$ , is a composite integer matrix. If  $\mathbf{N}$  is composite, it can be written as

$$\mathbf{N} = \mathbf{P} \mathbf{Q} \quad (3)$$

where  $\mathbf{P}$  and  $\mathbf{Q}$  are integer matrices such that  $|\det \mathbf{P}| > 2$  and  $|\det \mathbf{Q}| > 2$ . (As an aside it can be noted that  $\mathbf{N}$  is factorable whenever the absolute value of its determinant, which must be an integer, is not one or a prime number. Such a factorization is not unique, except possibly in the one-dimensional case).

The summation in (1) produces a distinct value for  $|\det \mathbf{N}|$  different values of  $\mathbf{k}$  and it is invertible if the region  $I_{\mathbf{N}}$  also contains  $|\det \mathbf{N}|$  samples. These  $|\det \mathbf{N}|$  values of  $\mathbf{n}$  and  $\mathbf{k}$  can be expressed as

$$\mathbf{k} = \mathbf{Q} \mathbf{m} + \mathbf{z} + \mathbf{N} \mathbf{r} \quad (4a)$$

$$\mathbf{n} = \mathbf{P} \mathbf{q} + \mathbf{p} + \mathbf{N} \mathbf{s} \quad (4b)$$

where  $\mathbf{p}$  and  $\mathbf{m}$  come from sets of integer vectors containing  $|\det \mathbf{P}|$  members and  $\mathbf{q}$  and  $\mathbf{z}$  come from sets of integer vectors containing  $|\det \mathbf{Q}|$  members. Substituting eqs. (4) into (1) reduces the

\*This work was supported, in part, by the National Science Foundation under grant ECS-7817201 and by the Joint Services Electronics Program under contract DAAG29-81-K-0024.

computation of a matrix- $N$  DFT into the computation of  $|\det P|$  matrix- $Q$  DFTs plus  $|\det Q|$  matrix- $P$  DFTs plus  $|\det N|$  additional complex multiplications. The net computational effort is less than if (1) is evaluated directly. If either  $P$  or  $Q$  is composite, a similar decomposition can be used to evaluate the smaller DFTs.

The two most common algorithms for evaluating the multidimensional rectangular DFT, the row-column algorithm and the vector-radix algorithm, correspond to special cases of this algorithm. Their specific relationship to the general algorithm is discussed in [2].

While the generalized Cooley-Tukey algorithm is elegant from a conceptual point of view, it is difficult to implement for non-diagonal periodicity matrices. The difficulty lies with the vector equivalent of the bit-reversal operation. A resolution of this difficulty is known for the hexagonal case [5], but the resulting algorithm requires that the data be reindexed at each decimation stage in the algorithm. The algorithm described in the next section reduces the amount of data shuffling required.

#### SMITH NORMAL FORM

When  $N$  is diagonal, a multidimensional DFT can be efficiently implemented using either the row-column algorithm or the vector-radix algorithm. If  $N$  is non-diagonal, we can develop similar algorithms if we first write it in Smith normal form

$$N = U D V \quad (5)$$

where  $D$  is an integer diagonal matrix and  $U$  and  $V$  are unimodular, i.e.,  $|\det U| = |\det V| = 1$  and  $U$  and  $V$  are integer matrices. This decomposition can be performed by executing elementary row and column operations on  $N$ .

Substituting eq. (5) into (1), the DFT summation can be written as

$$X(k) = \sum_{n \in I_N} x(n) \exp[-jk^T V^{-1} (2\pi D^{-1}) U^{-1} n]. \quad (6)$$

Now, if we define

$$\begin{aligned} \hat{n} &= U^{-1} n \\ \hat{k} &= V^{-T} k \end{aligned}$$

the DFT summation reduces to

$$\hat{X}(\hat{k}) = \sum_{n \in I_N} \hat{x}(\hat{n}) \exp[-j\hat{k}^T (2\pi D^{-1}) \hat{n}]. \quad (7)$$

This sum represents a matrix- $D$  DFT. Furthermore since  $U$  is unimodular  $\hat{n}$  and  $n$  define the same lattice. The sequence  $\hat{x}(\hat{n})$  is simply a reindexed version of  $x(n)$ . Similarly  $\hat{X}(\hat{k})$  is a reindexed version of  $X(k)$ . This decomposition provides the

following algorithm for evaluating a matrix- $N$  DFT:

#### Algorithm A:

1. Express  $N$  in Smith normal form as  $N = U D V$ .
2. Scramble the input array according to the relation  $\hat{n} = U^{-1} n$ .
3. Compute a DFT of the resulting array using a matrix- $D$  DFT. Since  $D$  is diagonal, this can be done using either a row-column DFT or a vector-radix FFT algorithm.
4. Unscramble the output sequence according to the relation  $k = V^T \hat{k}$ .

Observe that with this algorithm the multidimensional arrays need to be reindexed at most twice, once at the beginning of the algorithm and once at the end.

While the matrix  $D$  is an  $M \times M$  integer matrix, the matrix- $D$  DFT at step 3 of the algorithm is not necessarily an  $M$ -dimensional DFT. To illustrate this fact consider a two-dimensional matrix  $N$  DFT for which  $|\det N| = N_1 N_2$ . For some  $N$  the matrix  $D$  will assume the form

$$D = \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix}. \quad (8)$$

For other  $N$  the matrix  $D$  will assume the form

$$D = \begin{bmatrix} N_1 N_2 & 0 \\ 0 & 1 \end{bmatrix}. \quad (9)$$

Although the Smith normal form for a matrix is not unique, the form for its diagonal substitute normally is. (One exception to this statement is discussed below).

If  $D$  has the form of eq. (8), the DFT of step 3 of Algorithm A corresponds to an  $N_1 \times N_2$  two-dimensional rectangular DFT. If  $D$  has the form of eq. (9), this DFT is an  $N_1 \times N_2$ -point one-dimensional DFT. Since the two-dimensional transform can be computed more efficiently than a one-dimensional transform with the same number of points an  $N$  matrix whose diagonal substitute is of the form of eq. (8) is to be preferred over one of the form of eq. (9). If  $N_1$  and  $N_2$  are relatively prime then diagonal equivalents of either form exist. This fact was exploited by Good [6] whose prime factor algorithm represents an efficient algorithm for evaluating a 1-D DFT. The prime factor algorithm works by writing a 1-D DFT as a 2-D DFT whose periodicity matrix is of the form of (9). The data are then permuted into a form where the periodicity matrix has the form of eq. (8) which is then evaluated using a row-column two-dimensional DFT, with an attendant computational savings.

In a similar fashion, when  $N$  is  $M$ -dimensional, the dimensionality of the matrix-DFT may vary from 1 to  $M$ .

### MATRIX PRIME FACTOR ALGORITHM

With Algorithm A, the evaluation of an  $M$ -dimensional matrix- $N$  DFT can be accomplished by means of a rectangular DFT of dimensionality less than or equal to  $M$ . It can also be accomplished by using a higher dimensional rectangular DFT by using a generalization of Good's prime factor algorithm [6]. To explain the algorithm, however, we will need some results from lattice theory.

Let  $a_1, a_2, \dots, a_M$  be  $M$  linearly independent vectors in the  $M$ -dimensional real Euclidean space. The set of vectors

$$x = u_1 a_1 + \dots + u_M a_M \quad (10)$$

with integral  $u_1, \dots, u_M$  is called the lattice with basis  $a_1, \dots, a_M$ . If the vectors  $a_1, \dots, a_M$  are combined into a matrix,  $A$ , eq. (10) can be written as

$$x = A u. \quad (11)$$

Let the lattice generated by the matrix  $A$  be denoted  $L_A$ . There is a one-to-many relationship between lattices and matrices. To each nonsingular matrix  $A$  corresponds a lattice  $L_A$ , but to each lattice  $L_A$  there is a whole class of nonsingular matrices. Two matrices  $A$  and  $B$  belong to the same class if  $A = B U$  where  $U$  is a unimodular matrix.

If a lattice  $L_B$  is contained in a lattice  $L_A$ , then  $L_B$  is called a sublattice of  $L_A$ . In this case  $B = A C$  where  $C$  is an integer matrix. The set of vectors common to two lattices  $L_A$  and  $L_B$  constitute a lattice  $L_D$  called the greatest common sublattice of  $L_A$  and  $L_B$ .

If  $n$  and  $m$  are two vectors belonging to a lattice  $L_A$ , and if  $L_B$  is a sublattice of  $L_A$ , we will say that  $n$  is congruent to  $m$  modulo  $B$ , written

$$n \equiv m \pmod{B} \quad (12)$$

if  $(n-m)$  is a vector belonging to  $L_B$ . This relation defines a set of equivalence classes, called the set of residues modulo  $B$ , where a class  $[n]$  is

$$[n] = \{m \in L_A \text{ such that } m \equiv n \pmod{B}\} \quad (13)$$

This set of classes is denoted  $L_{A/B}$ .

Now we are ready to present a Chinese remainder theorem for integer vectors. Suppose that  $N$  is a composite integer matrix such that

$$N = P_1 Q_1 = Q_2 P_2$$

where  $|\det P_1| = |\det P_2| = p$ ,  $|\det Q_1| = |\det Q_2| = q$ , and  $p$  and  $q$  are relatively prime. Then  $L_{I/N}$  is isomorphic to  $L_{I/P_2} \times L_{I/Q_1}$ . Thus any integer vector  $k$  from the "region"  $L_{I/N}$  can be represented by the vector pair  $(k_1, k_2)$  where

$$\begin{aligned} k_1 &= k \pmod{P_2^T} \\ k_2 &= k \pmod{Q_1^T} \end{aligned} \quad (14)$$

The inverse mapping is given by

$$k = (A k_1 + B k_2) \pmod{N^T}$$

where

$$\begin{aligned} A k_1 &\equiv k_1 \pmod{P_2^T} \\ B k_2 &\equiv k_2 \pmod{Q_1^T} \end{aligned}$$

A second isomorphism is given by

$$n = (n_1, n_2)$$

where  $Q_2 n_1 + P_1 n_2 \equiv n \pmod{N}$ .

Substituting the two inverse relations into eq. (1), the DFT summation can be written.

$$X(k_1, k_2) = \sum_{n_1 \in L_{I/P_1}} \sum_{n_2 \in L_{I/Q_2}} x(n_1, n_2) \times e^{-j2\pi k_1^T P_2^{-1} n_1 - j2\pi k_2^T Q_1^{-1} n_2} \quad (12)$$

The resulting algorithm is similar to Algorithm A in that it involves shuffling the data, performing a DFT and then shuffling the result. The DFT formula in (12) reduces the computation to a number of smaller DFT's. A matrix- $Q$  DFT is evaluated for each value of the index  $n_1$  and then a matrix- $P$  DFT is evaluated for each value of the index  $k_2$ . The number of complex multiplications is then

$$m = |\det P| m_2 + |\det Q| m_1, \quad (13)$$

where  $m_1$  and  $m_2$  are the number of multiplications for a matrix- $P$  DFT and a matrix- $Q$  respectively.

While eq. (12) indicates the required computations, it is not clear that an efficient order-

ing for the data can be found. That task is made easier if a standard basis for each of the lattices and sublattices is used. With no loss of generality let us confine ourselves to the two-dimensional case and let us consider the evaluation of a 2-D matrix-P DFT of the form

$$X(\mathbf{k}) = \sum_{\mathbf{n} \in L_{I/P}} x(\mathbf{n}) \exp[-j2\pi \mathbf{k}^T \mathbf{P}^{-1} \mathbf{n}] \quad (14)$$

Let  $\mathbf{P} = [p_1, p_2]$ . Then it can be shown that there exist vectors  $\mathbf{x}_1$  and  $\mathbf{x}_2$  such that

$$\mathbf{x}_1 = h_{11} \mathbf{p}_1$$

$$\mathbf{x}_2 = h_{21} \mathbf{p}_1 + h_{22} \mathbf{p}_2$$

$$h_{11} > 0, \quad h_{22} > h_{21} > 0$$

and  $\mathbf{x}_1$  and  $\mathbf{x}_2$  form a basis for  $L_P$ . Inverting these equations gives

$$\mathbf{p}_1 = g_{11} \mathbf{x}_1$$

$$\mathbf{p}_2 = g_{21} \mathbf{x}_1 + g_{22} \mathbf{x}_2$$

Then the set of vectors

$$u_1 \mathbf{x}_1 + u_2 \mathbf{x}_2$$

for integer values of  $u_1$  and  $u_2$  in the range

$$0 < u_1 < g_{11}$$

$$0 < u_2 < g_{22}$$

constitute a representative system of residue classes for  $L_{I/P}$ . Similarly there exist vectors  $\mathbf{y}_1, \mathbf{y}_2$  and integers  $\ell_{11}, \ell_{22}$  such that the set of vectors

$$v_1 \mathbf{y}_1 + v_2 \mathbf{y}_2$$

$$0 < v_1 < \ell_{11}$$

$$0 < v_2 < \ell_{22}$$

constitute a representative set of residue classes for  $L_{I/N^T}$ . Thus if  $x(\mathbf{n}) = x(u_1, u_2)$  and  $X(\mathbf{k}) = X(v_1, v_2)$ , the DFT becomes

$$X(v_1, v_2) = \sum_{u_1=0}^{g_{11}-1} \sum_{u_2=0}^{g_{22}-1} x(u_1, u_2) \exp[-j(v_1 y_1 + v_2 y_2)^T \mathbf{P}^{-1} (u_1 \mathbf{x}_1 + u_2 \mathbf{x}_2)]$$

$$0 < v_1 < \ell_{11}$$

$$0 < v_2 < \ell_{22}$$

This DFT is now in the form of a DFT with a rectangular region of support.

## REFERENCES

- [1] R. M. Mersereau and T. C. Speake, "The processing of periodically sampled multidimensional signals," IEEE Trans. Acoustics, Speech, Signal Processing, v. ASSP-31, Feb. 1983.
- [2] R. M. Mersereau and T. C. Speake, "A unified treatment of Cooley-Tukey algorithms for the evaluation of the multidimensional DFT," IEEE Trans. Acoustics, Speech, Signal Processing, v. ASSP-29, No. 5, pp. 1011-1018, Oct. 1981.
- [3] R. M. Mersereau, "The processing of hexagonally sampled two-dimensional signals," Proc. IEEE, vol. 67, pp. 930-949, June 1979.
- [4] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," Math. Comput., vol. 19, no. 90, pp. 296-301, 1965.
- [5] P. K. Murphy and N. C. Gallagher, "Hexagonal sampling techniques applied to Fourier and Fresnel digital holograms," J. Opt. Soc. Amer., vol. 72, pp. 929-937, July 1982.
- [6] I. J. Good, "The interaction algorithm and practical Fourier series," J. Royal Stat. Soc., ser. B, vol. 20 (1958), pp. 361-372. Addendum, 22 (1960), pp. 372-375.