

分散形進化論的デジタルフィルタのFPGA上での 実現のためのハードウェア構成

Hardware Structure of Distributed Evolutionary Digital Filters for a Realization on an FPGA

A. S. ジャヤンタ セナラツ*, 阿部 正英*, 川又 政征*

A. S. Jayantha Senarath*, Masahide Abe*, Masayuki Kawamata*

*東北大学大学院工学研究科

*Graduate School of Engineering, Tohoku University

キーワード : 適応デジタルフィルタ (Adaptive Digital Filter), 進化論的計算手法 (Evolutionary Computation), 進化論的デジタルフィルタ (Evolutionary Digital Filter: EDF), EDF モジュール (EDF Module)

連絡先 : 〒980-8579 仙台市青葉区荒巻字青葉05 東北大学大学院工学研究科電子工学専攻川又研究室
A. S. ジャヤンタ セナラツ, Tel.: (022)217-7095, Fax: (022)263-9169, Email: jayantha@mk.ecei.tohoku.ac.jp

1. はじめに

著者らは, 進化論的計算手法による適応デジタルフィルタである進化論的デジタルフィルタ (Evolutionary Digital Filter: EDF) を提案している¹⁾. EDF には, (1) 多峰性の評価関数空間において極小値にとらわれずに最小値を探索できる, (2) 応用にあわせて評価関数を選択できる, (3) 極が単位円に近い場合でもフィルタの安定性を考慮する必要がない, という利点がある. しかし, EDF には, LMS アルゴリズムを用いた適応デジタルフィルタ (LMS-ADF) よりも, 1入力あたりの処理時間が長いという問題点がある. これは, EDF が多くの内部フィルタを持ち, これにより多点探索を行っているからである.

この問題を解決する方法として, 著者らは, 分散形 EDF を提案している²⁾. この方法でさらに

生物個体に対応する内部フィルタを並列に動作させることが容易になる. このため, ハードウェア化によるフィルタ単位の並列処理によっても1入力当りの処理時間を短縮することができる. また, EDF については, FPGA 上に実現することで, その性能を評価している³⁾.

そこで, 本稿では, まず, EDF の適応アルゴリズムの概略, EDF モジュールのハードウェア構成の概略を述べ, 分散形EDF のハードウェア構成を提案する. その上で, これを FPGA に実装したときの動作周波数とハードウェア量, サンプリング周波数について検討する.

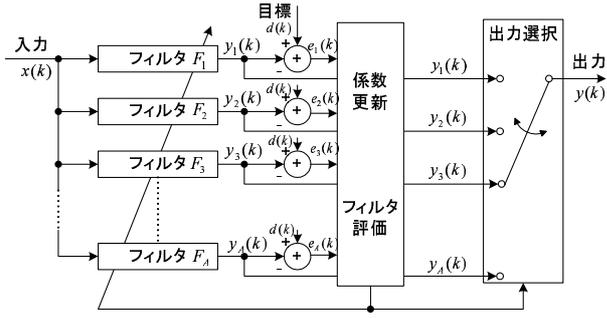


Fig. 1 EDF の構造

2. 進化論的デジタルフィルタ

2.1 進化論的デジタルフィルタの構造

進化論的デジタルフィルタ (EDF) の構造を図 1 に示す。EDF は係数可変のフィルタを内部に複数個持ち、それらの係数を進化論的計算手法に基づく適応アルゴリズムによって更新する。EDF の内部フィルタ F_i は、それぞれ同一の構成を持つ。また、それらの内部フィルタ F_i の構成は自由に変更が可能である。

2.2 適応アルゴリズム

EDF の適応アルゴリズムは (1) フィルタリング過程と (2) 適応過程に分けることができる。この節ではそれぞれを概説する。

2.2.1 フィルタリング過程

本稿では、EDF の内部フィルタの構造を直接形 II の IIR フィルタとする。内部フィルタ F_i の出力 $y_i(k)$ は、入力 $x(k)$ とフィルタ係数 $a_{n,i}(k)$ と $b_{m,i}(k)$ 、フィルタ係数の内部状態 $s_i(k)$ から次のように求められる。

$$s_i(k) = x(k) - \sum_{n=1}^N a_{n,i}(k) s_i(k-n) \quad (1)$$

$$y_i(k) = \sum_{m=0}^M b_{m,i}(k) s_i(k-m) \quad (2)$$

ここで、 k は時刻であり、 N と M は内部フィルタの次数である。EDF では、フィルタ係数と内部状態を 1 組にして生物個体として扱う。ここで、 $C = \max(M, N)$ とし、フィルタ係数ベクトル W_i と内部状態ベクトル S_i をそれぞれ、

$$W_i(k) = [a_{1,i}(k), a_{2,i}(k), \dots, a_{N,i}(k), b_{0,i}(k), b_{1,i}(k), \dots, b_{M,i}(k)] \quad (3)$$

$$S_i(k) = [s_i(k-1), \dots, s_i(k-C)] \quad (4)$$

とおく。

EDF では、内部フィルタ F_i の出力誤差 $e_i(k)$ を用いて各個体の適応度を計算する。時刻 k における出力誤差 $e_i(k)$ は目標信号 $d(k)$ と内部フィルタの出力信号 $y_i(k)$ により、

$$e_i(k) = d(k) - y_i(k) \quad (5)$$

となる。各内部フィルタの適応度 $f_i(k)$ は、誤差 $e_i(k)$ と評価期間 T_0 の統計量を用いて求められる。例えば次のように適応度 $f_i(k)$ を求めることができる。

$$f_i(k) = \left\{ \frac{1}{T_0} \sum_{j=0}^{T_0-1} e_i^2(k-j) \right\}^{-1} \quad (6)$$

ここで求められた適応度 $f_i(k)$ は、後述する個体の生殖方法を決める時と EDF の出力となる最も適応度の高い個体の出力を選択する時に用いられる。

2.2.2 適応過程

EDF の適応過程では、生殖と選択により係数可変フィルタの係数を更新する。係数更新する方法には、無性生殖と有性生殖の 2 つがある。EDF では、個体を適応度の高さで 2 分し、適応度の高い個体は無性生殖を行い、適応度の低い個体は有性生殖を行う。

適応度の高い個体は、自分の近傍に子を生成する無性生殖を行う。これは、適応度の高い個体は、評価関数空間における最適値の近傍にあると考えられるためである。無性生殖では N_{ap} 個体の親 P_{ap}

が、それぞれ N_{ac} 個の子 P_{ac} を生成する。また、1 個体の親と生成される子の中で、最も適応度の高い個体のみが次世代に引き継がれる。無性生殖は以下の式で実現される。

$$W_{ac,i,j} = W_{ap,i} + r \cdot n_{i,j} \quad (7)$$

$$S_{ac,i,j} = S_{ap,i} \quad (8)$$

ここで、 $i = 1, 2, \dots, N_{ap}$ 、 $j = 1, 2, \dots, N_{ac}$ とする。また、 r は転写ミスの大きさであり、 $n_{i,j}$ は平均 0、分散 1 の正規乱数ベクトルである。生成された子は親と似た遺伝子を持つため、内部状態 S はそのまま引き継がれる。

適応度の低い個体は、有性生殖を行う。有性生殖では、 N_{sp} 個体の親 P_{sp} が 2 個体 1 組になり、それぞれ 1 個体の子 P_{sc} を生成する。そして、生成された子と親 2 個体の中で適応度の高い方の個体が次世代に引き継がれる。有性生殖は以下のような式で実現される。

$$W_{sc,m} = \frac{1}{2} (W_{sp,k(m)} + W_{sp,l(m)}) + s \cdot n_m \quad (9)$$

$$S_{sc,m} = 0 \quad (10)$$

ここで、 $k(m)$ と $l(m)$ は、 $\{1, 2, \dots, N_{sp}\}$ の中から重複なく選ばれるものとし、 $m = 1, 2, \dots, N_{sp}/2$ とする。また s は転写ミスの大きさであり、 n_m は平均 0、分散 1 の正規乱数ベクトルである。生成された子の遺伝子は親の遺伝子とは掛け離れてしまうため内部状態 S は 0 に設定される。

3. 分散形進化論的デジタルフィルタ

EDF の問題点である 1 入力当りの処理時間が長いという問題を解決することと、内部フィルタを並列に動作させることを目標にして、分散形 EDF を提案している。この節では、分散形 EDF の概要を述べる。

分散形 EDF では、生物群を分割することによりフィルタリングと係数更新を並列化する。この

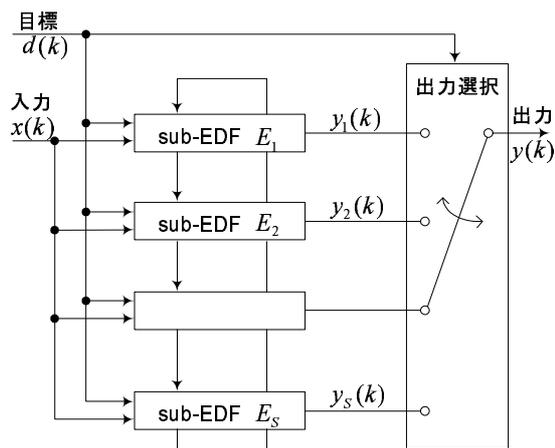


Fig. 2 分散形 EDF の構造

分割した生物群に EDF の構造と適応アルゴリズムを適用するために、分散形遺伝的アルゴリズム (Distributed Genetic Algorithm: DGA) の分割方法を用いている⁴⁾。図 2 は、分散形 EDF のブロック図である。

分散形 EDF は、複数の sub-EDF と出力選択部からなる。各 sub-EDF には、同一の入力 $x(n)$ を与える。出力選択部では、もっとも適応度の高い sub-EDF の出力を全体の出力 $y(n)$ として選択する。また分散形 EDF では、各 sub-EDF をリング状に結合し、隣接する sub-EDF 間で個体を交換する。

まず、同じサイズの S 個の小さな生物群にもとの EDF の生物群を分割する。各 sub-EDF は、分割された生物群を 1 つ持ち、生物数が少なくなった点を除いて、元の EDF と同じ処理を行う。また、ある時間間隔で各 sub-EDF の個体を交換する。

分散形 EDF のアルゴリズムは以下のようになる。

while 入力信号がある。

Step 1. T_m 世代間、各 sub-EDF で生殖、評価、選択・淘汰を行う。

Step 2. すべての sub-EDF に対して以下の手続きを行う。

Step 2-1. sub-EDF E_i の個体の中で

Table 1 EDF と分散形 EDF の個体数の比較

	EDF	sub-EDF	分散形 EDF
無性生殖における親個体の数	N_{ap}	N_{ap}/S	N_{ap}
無性生殖を行う親 1 個体が生成する子孫の数	N_{ac}	N_{ac}	N_{ac}
有性生殖における親の数	N_{sp}	N_{sp}/S	N_{sp}

Table 2 EDF モジュール version 2 の仕様

N_{ap}	32	無性生殖を行う親の数
N_{ac}	32	無性生殖を行う1匹の親が生成する子孫の数
r	0.1	無性生殖の転写ミスの大きさ
N_{sp}	32	有性生殖を行う親の数
s	0.5	有性生殖の転写ミスの大きさ
T_0	10	評価期間
S	4	生物群の分割数 (sub-EDF の数)
T_m	1	個体交換する世代の間隔
N_m	1	sub-EDF 間で交換する個体数

適応度の高い順に N_m 個体を隣接する sub-EDF E_{i+1} にコピーする。

Step 2-2. sub-EDF E_{i+1} では, N_m 個体をランダムに選択し, E_i からコピーされた個体を置き換える。

end

各 sub-EDF では, 無性生殖と有性生殖を行う親の数は, それぞれ N_{ap}/S と N_{sp}/S となる。これによって元の EDF と分散形 EDF の適応度評価される個体の総数は等しくなる。よって, 各 sub-EDF で適応度評価される個体の総数 A_s は次式で表される。

$$A_s = \frac{N_{ap}}{S} + \frac{N_{ap}}{S} N_{ac} + \frac{N_{sp}}{S} + \frac{1}{2} \frac{N_{sp}}{S} \quad (11)$$

表1に, 評価される個体数の詳細を示す。

以降で述べる EDF, sub-EDF, 分散形 EDF ハードウェアモジュールでは上記のパラメータを表2のように設定した。

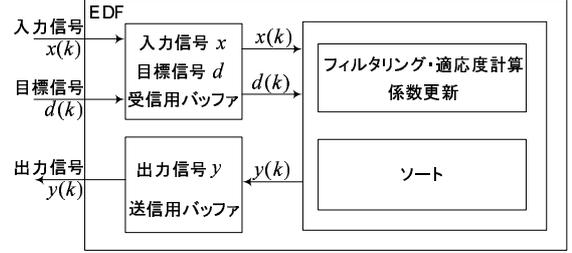


Fig. 3 EDF モジュール version 2 のハードウェア構成

4. EDF モジュール version 2 のハードウェア構成

4.1 ハードウェア構成

本節では, 著者らが提案している EDF のハードウェアモジュールである EDF version 2³⁾ のハードウェア構成を概説する。

EDF モジュール version 2 のハードウェア構成を図3に示す。EDF モジュール version 2 は, フィルタリング・適応度計算と係数更新を実現するモジュールとソートモジュールの2つの下位モジュールを持つ。表3に EDF モジュール version 2 の仕様を示す。小数点表示方式は, ハードウェア記述の容易さと EDF で扱うフィルタの係数の範囲を考慮して固定小数点の Q14 フォーマットに設定されている。ここで, Q14 フォーマットとは2進小数点以下の数字の個数が14個となるように小数点が置かれるフォーマットをさす。

Table 3 EDF モジュール version 2 の仕様

EDF	小数点表示方式	固定小数点 (Q14)
	データビット幅	16 bits
	個体数	$N_{ap} + N_{sp} \leq 64$
	フィルタ次数	$N \leq 3, M \leq 2$
メモリ	個体メモリ	512×16 bits , 256×16 bits

4.2 EDF モジュール version 2 の特徴

EDF モジュール version 2 のフィルタリング・適応度計算処理はステートマシン制御方式である。これは、その前の EDF モジュール version 1 のアセンブリ言語制御方式から変更された。これによって、処理クロック数削減と回路規模削減を実現している³⁾。しかし、ステートマシン制御方式に変更したことによりフィルタ構造と誤差関数の変更は回路合成を行わない限り不可能になる。これは、アセンブリ言語を使わないため汎用性が減少したからである。EDF モジュール version 2 ではフィルタ構造を直接形 II の IIR フィルタ、誤差関数を 2 乗平均誤差関数に固定する。そして、制御方式をステートマシン制御に変更し、効率よく処理することにより、実際の計算以外の余計な処理クロック数を削減する。

5. 分散形 EDF のハードウェア構成

5.1 ハードウェア構成

分散形 EDF のハードウェア構成を図4に示す。今回設計した分散形 EDF モジュールでは、sub-EDF の数 S を4にした。それぞれの sub-EDF モジュールは、前節で説明した EDF モジュール version 2 である。ここでは EDF モジュール version 2 をそのまま使うのではなく、モジュール間の通信ができるように migrator モジュールを埋め込んで使用

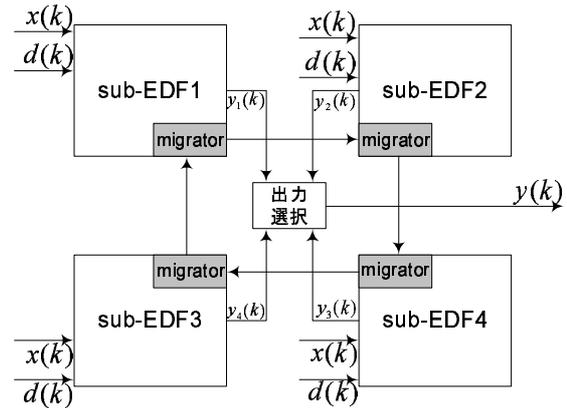


Fig. 4 分散形 EDF のハードウェア構成

する。この migrator モジュールによって sub-EDF 間の個体の交換が行われる。分散形 EDF では並列化の容易性と通信量を考慮して sub-EDF をリング状に結合させて個体交換を行う。

それぞれの sub-EDF の出力信号 $y_i(k)$ とそれぞれの出力の適応度 f_i は出力選択モジュールに入力される。その中から最も高い適応度 f_i に対する sub-EDF の出力信号 $y_i(k)$ を分散形 EDF の出力として選択して出力する。

5.2 sub-EDF 間の通信

sub-EDF モジュール間の通信過程のフローチャートを図5に示す。ここでフローチャートの2番目のプロセスで隣接する sub-EDF に個体交換開始信号を送信してから状態が変わる。それと同時に send レジスタの内容を常に出力し続ける。状態が変わって隣接する sub-EDF から個体情報を受け取る状態になって、新しい個体を take レジスタで受け取る。このような操作は sub-EDF モジュールがすべて同期を取って同じクロックで動作しているために可能になる。この方法によって sub-EDF 間の通信を行う部分が非常に少ないハードウェア資源で実現可能になる。

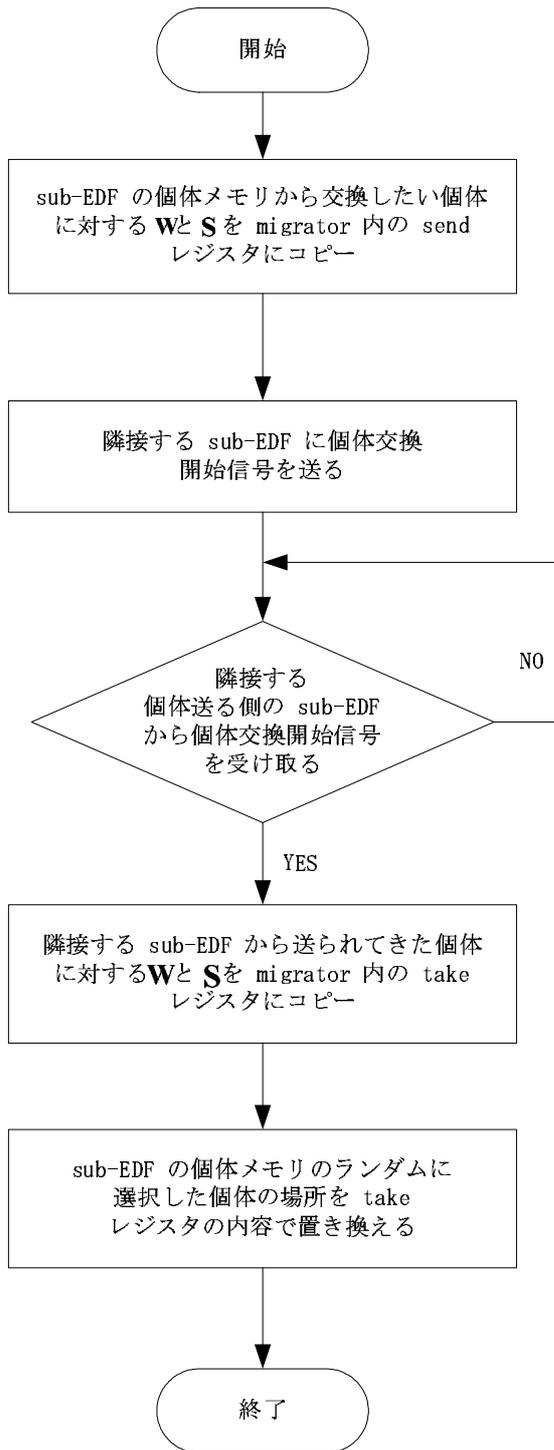


Fig. 5 個体交換過程のフローチャート

6. 論理合成と配置配線結果

本節で, migrator モジュールと sub-EDF モジュール, 分散形 EDF モジュールの論理合成結果と配置配線結果を示す. 論理合成と配置配線は表4の環

Table 4 FPGA 実装環境

使用言語	Verilog-HDL
論理合成, 配置配線ツール	Xilinx 社 Foundation ISE 6.2i
Target Device	Xilinx 社 Virtex-II (XC2V3000-4FG676)

Table 5 migrator モジュールの論理合成結果

migrator モジュール	論理合成
回路規模	29slices
動作周波数	264.2MHz

境で行った.

6.1 migrator モジュール

migrator モジュールは分散形 EDF モジュールを構成する sub-EDF の中に埋め込まれる. このモジュールは上記で説明した通りの過程で動作する. このモジュールの論理合成結果は表5のようになる.

6.2 sub-EDF モジュール

sub-EDF の構造は EDF version 2 とする. sub-EDF 間の個体交換はモジュールの中に埋め込まれる migrator モジュールによって行われる. そのようにして構成された sub-EDF モジュールの論理合成結果を表6に示す. EDF version 2 と比較した結果も一緒に示す. 表6より, 回路規模が減少し, 動作周波数は同じであることがわかる. 回路規模が減少するのは sub-EDF の内部フィルタの数(個体数)が減少するからである. また, 動作周波数が同じになっているのは, 回路のクリティカルパスが変わらないためと考えられる.

Table 6 sub-EDF モジュールの論理合成結果

モジュール	sub-EDF	EDF version 2
回路規模	1286slices	1351slices
動作周波数	76.7MHz	76.7MHz

6.3 分散形 EDF モジュール

図2に示したように sub-EDF モジュールを4つ組み合わせて分散形 EDF モジュールとする。その時、それぞれの sub-EDF の出力を出力選択モジュールに入力させ、もっとも適応度の高い sub-EDF 出力を分散形 EDF の出力として選択する。この分散形 EDF モジュールの論理合成結果は表7のようになる。ここで、EDF を一つのモジュールとして設計したときの場合 (EDF モジュール version 2) と比較する。

表7より、分散形 EDF の場合、EDF version 2 モジュールより少し小さい sub-EDF を4つ並列に動作しているために回路規模は EDF version 2 の回路規模の3.7倍になっていることが確認できる。

分散形 EDF モジュールの動作周波数は表6で示した sub-EDF モジュールと同じになる。それは、sub-EDF モジュールを4つ並列に動作させるので、クリティカルパスは1個の sub-EDF と同じになるためである。

表8に分散形 EDF と sub-EDF , EDF version 2 の1個体が入力信号1サンプルを処理するために必要なクロック数を示す。表8より、どのモジュールもフィルタリング・適応度計算と係数更新に同じクロック数が必要になることがわかる。これは、一つの個体のフィルタリング・適応度計算と係数更新の過程はどのモジュールでも同じとなるからである。しかし、ソートに必要なクロックは個体数に比例しているため、モジュール内の個体数が減少するとクロック数も減少する。

Table 7 分散形 EDF モジュールの論理合成結果

モジュール	分散形 EDF	EDF version 2
回路規模	4964slices	1351slices
動作周波数	76.7MHz	76.7MHz

Table 9 分散形 EDF と EDF version 2 モジュールの配置配線結果

モジュール	分散形 EDF	EDF version 2
回路規模	5479slices	1512slices
動作周波数	76.7MHz	76.7MHz
サンプリング周波数	23.7kHz	5.8kHz

6.4 性能評価

表9に分散形 EDF と EDF モジュール version 2 の配置配線結果を示す。使用した FPGA とツールは表4のとおりである。また、分散形 EDF と EDF モジュール version 2 の1個体が入力信号1サンプルを処理するのに必要なクロック数について表8に示した。これらの表から、sub-EDF モジュールを4並列にした分散形 EDF のサンプリング周波数は 23.7kHz となり、4.1 倍に性能が向上することがわかった。また、表7より回路規模は 3.6 倍になることもわかった。

7. まとめ

本論文では、著者らがこれまでに提案した EDF モジュール version 2 のハードウェア構成を利用し、分散形 EDF のハードウェア構成を提案した。そして、提案するハードウェア構成を用いた分散形 EDF モジュールを FPGA 上に実装し、回路規模と動作周波数を評価した。

Table 8 sub-EDF と分散形 EDF の 1 個体が入力信号 1 サンプルを処理するのに必要なクロック数

モジュール	分散形 EDF	sub-EDF	EDF version 2
フィルタリング・ 適応度計算	9.6	9.6	9.6
係数更新	2.0	2.0	2.0
ソート	0.09	0.09	0.4
個体交換	0.03	-	-

参考文献

- 1) M. Abe and M. Kawamata, "Evolutionary digital filtering for IIR digital filters based on the cloning and mating reproduction," IEICE Trans. Fundamentals, vol. E81-A, no. 3, March 1998.
- 2) M. Abe and M. Kawamata, "Distributed evolutionary digital filters for IIR adaptive digital filters," IEICE Trans. Fundamentals, vol. E84-A, no. 8, August 2001.
- 3) 阿部正英, 荒井拓貴, 川又政征, "進化論的デジタルフィルタのハードウェア実現のための構成とその FPGA 上での実現," 第17回 回路とシステム, 軽井沢ワークショップ, pp. 117-122, April 2004.
- 4) R. Tanese, "Distributed genetic algorithms," Proc. Third International Conference on Genetic Algorithm, June 1989.