

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4281419号
(P4281419)

(45) 発行日 平成21年6月17日(2009.6.17)

(24) 登録日 平成21年3月27日(2009.3.27)

(51) Int.Cl. F I
H03F 1/32 (2006.01) H03F 1/32

請求項の数 4 (全 39 頁)

(21) 出願番号	特願2003-157249 (P2003-157249)	(73) 特許権者	000002130
(22) 出願日	平成15年6月2日(2003.6.2)		住友電気工業株式会社
(65) 公開番号	特開2004-363733 (P2004-363733A)		大阪府大阪市中央区北浜四丁目5番33号
(43) 公開日	平成16年12月24日(2004.12.24)	(74) 代理人	100064746
審査請求日	平成18年2月23日(2006.2.23)		弁理士 深見 久郎
		(74) 代理人	100085132
			弁理士 森田 俊雄
		(74) 代理人	100083703
			弁理士 仲村 義平
		(74) 代理人	100096781
			弁理士 堀井 豊
		(74) 代理人	100098316
			弁理士 野田 久登
		(74) 代理人	100109162
			弁理士 酒井 将行

最終頁に続く

(54) 【発明の名称】 前置型非線形歪補償器

(57) 【特許請求の範囲】

【請求項1】

信号変換器の逆特性を用いて、入力信号を変換し、変換後の信号を信号変換器に供給する変換用サポートベクトルマシンと、

前記信号変換器で変換された変換出力信号を受け、前記変換出力信号列により構成される信号ベクトルと、過去に抽出された変換出力信号列により構成される信号ベクトルとに基づいて、前記変換用サポートベクトルマシンにおける変換後の入力信号を、第1の不感帯を有する損失関数に基づいて推定する推定用サポートベクトルマシンと、

前記変換用サポートベクトルマシンおよび前記推定用サポートベクトルマシンに接続され、前記変換用サポートベクトルマシンの出力および前記推定用サポートベクトルマシンの出力に基づいて、前記変換用サポートベクトルマシンおよび前記推定用サポートベクトルマシンの係数を更新するための係数更新手段と、

前記入力信号としきい値とを比較し、前記入力信号が前記しきい値よりも大きい場合、当該入力信号を含むデータセットをサポートベクトルとして採用しないようにするデータ選定手段とを備え、

前記データ選定手段は、過去の所定数の前記入力信号の絶対値に基づいて、前記しきい値を設定する、前置型非線形歪補償器。

【請求項2】

前記係数更新手段は、

前記変換出力信号を受け、新たに受けた前記変換出力信号の信号列により構成されるべ

クトルであって、かつ、第2の不感帯の外にあるベクトルをサポートベクトルの候補として抽出し、既にサポートベクトルとして抽出されているベクトルの集合に加え、前記ベクトルの集合を新たにサポートベクトルの集合とみなし、前記ベクトルの集合の中から、前記第1の不感帯の外にあるベクトルをサポートベクトルとして再抽出する抽出手段と、

前記抽出されたサポートベクトルに対する係数を計算する計算手段とを有する、請求項1記載の前置型非線形歪補償器。

【請求項3】

前記データ選定手段は、所定の期間内の過去の所定数の前記入力信号の絶対値に基づいて、前記しきい値を更新する、請求項1記載の前置型非線形歪補償器。

【請求項4】

前記信号変換器は、増幅器である、請求項1～3のいずれかに記載の前置型非線形歪補償器。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、前置型非線形歪補償器に関し、特に、サポートベクトルマシン（以下「SVM」という）を回帰分析に用いた前置型非線形歪補償器に関する。

【0002】

【従来の技術】

近年、通信における非線形デバイス（以下「NLD」という）の非線形歪を補償する技術として、前置型非線形歪補償器が注目されている。代表的な前置型非線形歪補償器として、ルックアップテーブルを用いたものと、NLDの逆特性を推定するものがある。ここで、NLDの典型的なものとしては、たとえば、増幅器が挙げられる。

【0003】

ルックアップテーブルを用いた前置型非線形歪補償器は、増幅器の特性を入力信号の振幅値に対して微小区間に分け、その区間の代表値をテーブルに格納しておき、送信信号とそれに対応するテーブルの補償値を乗算して出力信号とするものである。また、前置型非線形歪補償器入力とNLD出力を比較し、その結果から適応的にテーブル内の補償値を更新する前置型非線形歪補償器も既に開発されている。

【0004】

一方、増幅器の逆関数を推定する前置型非線形歪補償器として、`pth-order-predistorter`と呼ばれるものがある。この前置型非線形歪補償器は、信号変換器の逆関数を複素多項式近似するものである。複素多項式は、非線形関数の表現形式として最も一般的なポルテラ級数より導出され、歪補償が行なわれる。また、前置型非線形歪補償器入力とNLD出力を比較し、その結果から適応的に複素多項式係数を更新する前置型非線形歪補償器も既に開発されている。

【0005】

しかし、ルックアップテーブルを用いた前置型非線形歪補償器では、テーブル内の値は離散値をとるため、値の不連続性により、フロアノイズが上昇する。フロアノイズを解消するためには、データ数を増やす必要があるが、データ数を増やすとメモリ容量が多くなるという問題がある。

【0006】

`pth-order-predistorter`では、計算精度を上げるために複素多項式の次数を増やす必要がある。このため、計算時間がかかり、解が収束しにくいという問題がある。

【0007】

さらに、広帯域信号に対しては、ルックアップテーブル、`pth-order-predistorter`ともに、信号帯域内の各周波数ごとに振幅、位相の非線形性を補償する必要があるため、ハードウェア化が困難であるという問題がある。

【0008】

10

20

30

40

50

ところで、近年、入出力観測データからプロセスのモデリングを行うという問題は、非線形回帰問題やシステム同定などの多くの分野において研究の目的となっている。上記前置型非線形歪補償器の最適な特性を得るという問題も、ひずみを補償する対象となる増幅器を未知の非線形システムとみなせば、この補償する対象となる増幅器の特性の逆関数のモデリングの問題に帰着される。

【 0 0 0 9 】

このようなプロセスモデリングの問題に取り組む方法は、大きく分けてグローバルモデリングとローカルモデリングの2つに分けることができる。

【 0 0 1 0 】

ローカルモデリングは、重み平均法、局所重み回帰法やデータベース学習であるJust-in-time法などがある。

【 0 0 1 1 】

これに対して、グローバルモデリングは、データベースの全てに適合する一つの関数を作る手法である。このような関数の作成は、ニューラルネットワークモデリングや非線形統計回帰の他の方法で使われている手法である。

【 0 0 1 2 】

このようなグローバルモデリングに対する統計的学習理論は、1960年代から存在していたが、実用的なツールとして用いられるようになったのは、1990年半ば以降にV.N. Vapnik等によって、サポートベクトルマシンが提案されたことを契機とする(たとえば、非特許文献1~3を参照)。

【 0 0 1 3 】

このようなサポートベクトルマシンを用いた統計的学習理論によるプロセスモデリングには、一般的に、以下のような特徴がある。

【 0 0 1 4 】

(1) 他の手法に比べると、収束速度に勝る。

(2) 常に唯一の解を得ることができる。

【 0 0 1 5 】

(3) さまざまな学習問題に容易に適用できる。

(4) 問題の複雑さを問題の次元とは独立であるサポートベクトルの個数によって特徴づけているために、いわゆる「次元の呪い (curse of dimensionality)」から逃れることができる。

【 0 0 1 6 】

さらに、上述したサポートベクトルマシンは、回帰問題の学習に用いることのできるサポートベクトル回帰 (Support Vector Regression (以下、「SVR」と呼ぶ)) に拡張されている。このようなSVRについては、非特許文献4に開示がある。

【 0 0 1 7 】

さらに、このようなサポートベクトルマシンを上述した前置型非線形歪補償器に適用する試みが、非特許文献5に開示されている。

【 0 0 1 8 】

【非特許文献1】

V.N.Vapnik : The Nature of Statistical Learning Theory, Springer-Verlag, New York(1995)

【 0 0 1 9 】

【非特許文献2】

V.N.Vapnik : Statistical Learning Theory, J.Willey & Sons, New York(1998)

【 0 0 2 0 】

【非特許文献3】

B.Scholkopf et al.ed, "Advances in Kernal Methods", The MIT Press, 1999.

【 0 0 2 1 】

【非特許文献4】

10

20

30

40

50

N.Cristianini, J.Shave-Taylor "An Introduction to Support Vector Machines", Cambridge University Press, 2000.

【 0 0 2 2 】

【非特許文献 5】

伊藤隆徳, 大森浩充, 佐野昭 B - 5 - 1 2 4 「サポートベクトル法を用いた適応非線形歪補償」2001年電子情報通信学会総合大会予稿集、p. 522, 2001年

【 0 0 2 3 】

【発明が解決しようとする課題】

しかしながら、サポートベクトルマシンをデジタル信号受信装置の前置型非線形歪補償器に用いる場合には、以下のような点を考慮する必要がある。

10

【 0 0 2 4 】

ここで、通信に利用される N L D のうち、最も伝送特性に影響を及ぼすのは増幅器である。

【 0 0 2 5 】

図 20 は、一般的な増幅器の入出力特性を示す図である。

図 20 に示すように、入力信号が小さい領域では線形的に動作するが、入力信号が大きくなるにつれて出力信号は飽和する。

【 0 0 2 6 】

通常は、このような増幅器の非線形歪による影響を軽減するために、増幅器に対してマージンを設定し、その分だけ増幅器への入力レベルを下げて動作させる。このマージンを「バックオフ」と呼ぶ。

20

【 0 0 2 7 】

所要バックオフの大きさは変調方式に依存する。たとえば、変調方式が $\pi/4$ シフト Q P S K (Quadrature Phase Shift Keying) であれば、振幅変動が小さいのでバックオフは小さくてもよい。しかしながら、たとえば変調方式が直交周波数分割多重 (以下「O F D M (Orthogonal Frequency Division Multiplexing)」という) であれば、振幅変動が大きいのでバックオフを大きく設定する必要がある。

【 0 0 2 8 】

ところが、バックオフを大きく設定すると、増幅器の電力効率が大幅に劣化してしまう。そこで、信号のような振幅変動が大きい信号に対しても、なるべく小さいバックオフで動作させるのが望ましい。

30

【 0 0 2 9 】

しかしながら、バックオフを小さく設定すると、S V R のようなグローバルモデリングを用いた前置型非線形歪補償器では、補償できないほどに過剰に大きな歪を有する信号が、前置型非線形歪補償器に入力されることになる。

【 0 0 3 0 】

ここで、増幅器の逆特性、すなわち理想的に動作する前置型非線形歪補償器の特性は、図 20 に示すとおりである。ここで、前置型非線形歪補償器の出力は、入力が多い場合に非常に大きくなってしまふ。これは同時に、また増幅器への入力自体が非常に大きくなることを意味する。

40

【 0 0 3 1 】

ただし、このように大きな入力が仮に増幅器に入力されても、増幅器の特性自体が飽和しているのもはや、前置型非線形歪補償器では、補償を行なうことができない。この場合、特に、S V R のように補償動作を学習するようなシステムでは、有効な学習ができなくなるという問題が生じる。

【 0 0 3 2 】

本発明は、上述の課題を解決するためになされたもので、その目的は、振幅変動が大きい信号に対してもなるべく小さいバックオフで動作可能であり、なおかつ高精度で歪補償を可能とする前置型非線形歪補償器を提供することである。

【 0 0 3 3 】

50

【課題を解決するための手段】

本発明のある局面に従う前置型非線形歪補償器は、信号変換器の逆特性を用いて、入力信号を変換し、変換後の信号を信号変換器に供給する変換用サポートベクトルマシンと、信号変換器で変換された変換出力信号を受け、変換出力信号列により構成される信号ベクトルと、過去に抽出された変換出力信号列により構成される信号ベクトルとに基づいて、変換用サポートベクトルマシンにおける変換後の入力信号を、第1の不感帯を有する損失関数に基づいて推定する推定用サポートベクトルマシンと、変換用サポートベクトルマシンおよび推定用サポートベクトルマシンに接続され、変換用サポートベクトルマシンの出力および推定用サポートベクトルマシンの出力に基づいて、変換用サポートベクトルマシンおよび推定用サポートベクトルマシンの係数を更新するための係数更新手段と、入力信号としきい値とを比較し、入力信号がしきい値よりも大きい場合、当該入力信号を含むデータセットをサポートベクトルとして採用しないようにするデータ選定手段とを備える。

10

【0034】

好ましくは、係数更新手段は、変換出力信号を受け、新たに受けた変換出力信号の信号列により構成されるベクトルであって、かつ、第2の不感帯の外にあるベクトルをサポートベクトルの候補として抽出し、既にサポートベクトルとして抽出されているベクトルの集合に加え、前記ベクトルの集合を新たにサポートベクトルの集合とみなし、前記ベクトルの集合の中から、前記第1の不感帯の外にあるベクトルをサポートベクトルとして再抽出する抽出手段と、前記抽出されたサポートベクトルに対する係数を計算する計算手段とを有する。

20

【0035】

好ましくは、前記しきい値は、所定の定数である。

好ましくは、前記データ選定手段は、過去の所定数の入力信号の絶対値に基づいて、しきい値を設定する。

【0036】

好ましくは、データ選定手段は、所定の期間内の過去の所定数の入力信号の絶対値に基づいて、しきい値を更新する。

【0037】

好ましくは、信号変換器は、増幅器である。

【0038】

【発明の実施の形態】

以下、図面を参照しつつ本発明の実施の形態について詳細に説明する。以下の説明では、同一の構成部分には同一の符号を付してあり、それらの名称および機能も同じである。したがって、それらについての詳細な説明は繰り返さない。

【0039】

[サポートベクトルマシンによる回帰問題の学習の一般論]

以下、本発明の説明を行うにあたり、その前提となるサポートベクトルマシンによる回帰問題の学習について、まとめておく。

【0040】

1 - 1 . 線形回帰問題の学習

40

1 - 1 - 1 線形回帰問題

まず、サポートベクトルマシン(SVM)の考えに基づき、サポートベクトル回帰(Support Vector Regression: SVR)の立場から線形回帰問題を解く方法について述べる。この問題は、以下のように定式化できる。

【0041】

[問題]

m入力1出力の未知線形回帰 $f: x \rightarrow y$ があり、トレーニングデータ

$(x_1, y_1), \dots, (x_n, y_n)$ $x_i \in \mathbb{R}^m, y_i \in \mathbb{R}$ (1.1)

が与えられているとする。このとき、トレーニングデータを用いてこの線形回帰を近似する関数

50

$$f(x, w) = w^T x \quad (1.2)$$

を求める。

【 0 0 4 2 】

1 - 1 - 2 損失関数

ここで、線形 不感損失関数 (Linear -Insensitive Loss Function) $L(x, y, f(x, w))$ ならびに二次 不感損失関数 (Quadratic -Insensitive Loss Function) $L_2(x, y, f(x, w))$ を次のように定義する。

【 0 0 4 3 】

【数 1】

$$\begin{aligned} L^\varepsilon(x, y, f(x, w)) &= |y - f(x, w)|_\varepsilon \\ &= \max(0, |y - f(x, w)| - \varepsilon) \end{aligned} \quad (1.3)$$

10

$$\begin{aligned} L_2^\varepsilon(x, y, f(x, w)) &= |y - f(x, w)|_\varepsilon^2 \\ &= (\max(0, |y - f(x, w)| - \varepsilon))^2 \end{aligned} \quad (1.4)$$

図 1 は、このような線形 不感損失関数および二次 不感損失関数を示す図である。図 1 (a) は、線形 不感損失関数であり、図 1 (b) は、二次 不感損失関数を示す。

【 0 0 4 4 】

20

1 - 1 - 3 二次 不感損失関数による回帰問題

以下では、二次 不感損失を用いた場合のサポートベクトル回帰について考える。サポートベクトルマシンに基づいた主問題は、以下のように表現できる。

【 0 0 4 5 】

【数 2】

$$\min_w \|w\|^2 + C \sum_{i=1}^n (\max(0, |y_i - f(x_i, w)| - \varepsilon))^2 \quad (1.5)$$

この主題は、次のように書き換えることができる。

30

【 0 0 4 6 】

【数 3】

スラック変数 $\xi_i, \hat{\xi}_i (\geq 0)$ 等を導入すると、

$$\min_w \|w\|^2 + C \sum_{i=1}^n (\xi_i^2 + \hat{\xi}_i^2) \quad (1.6)$$

$$\text{subject to} \quad -y_i + w^T x_i \leq \varepsilon + \xi_i \quad i = 1, \dots, n \quad (1.7)$$

$$y_i - w^T x_i \leq \varepsilon + \hat{\xi}_i \quad i = 1, \dots, n \quad (1.8)$$

$$\hat{\xi}_i \geq 0 \quad \xi_i \geq 0 \quad i = 1, \dots, n \quad (1.9)$$

40

ただし、 C はモデルの複雑さと制約条件の緩和 (トレーニング誤差) との間のトレードオフを設定する正則化パラメータである。

【 0 0 4 7 】

【数 4】

ラグランジュ乗数 $\alpha_i, \hat{\alpha}_i \geq 0$ を導入すると、この問題のラグランジュ関数は以下のようになる。

$$L = \frac{1}{2} \|\mathbf{w}\|^2 + \frac{C}{2} \sum_{i=1}^n (\xi_i^2 + \hat{\xi}_i^2) + \sum_{i=1}^n \alpha_i (\mathbf{w}^T \mathbf{x}_i - y_i - \varepsilon - \xi_i) + \sum_{i=1}^n \hat{\alpha}_i (y_i - \mathbf{w}^T \mathbf{x}_i - \varepsilon - \hat{\xi}_i)$$

subject to $\alpha_i, \hat{\alpha}_i \geq 0 \quad i = 1, \dots, n$

(1.10)

式 (1.10) に対する Karush-Kuhn-Tucker (KKT) 条件により、以下の関係が成り立つ。

【 0 0 4 8 】

【 数 5 】

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} + \sum_{i=1}^n (\alpha_i - \hat{\alpha}_i) \mathbf{x}_i = 0 \quad i = 1, \dots, n \quad (1.11)$$

20

$$\frac{\partial L}{\partial \xi_i} = C \xi_i - \alpha_i = 0 \quad i = 1, \dots, n \quad (1.12)$$

$$\frac{\partial L}{\partial \hat{\xi}_i} = C \hat{\xi}_i - \hat{\alpha}_i = 0 \quad i = 1, \dots, n \quad (1.13)$$

これらの関係式を式 (1.10) に代入すると次の双村問題に書き換えられる。

【 0 0 4 9 】

【 数 6 】

$$\max_{\alpha} \sum_{i=1}^n y_i (\hat{\alpha}_i - \alpha_i) - \varepsilon \sum_{i=1}^n (\hat{\alpha}_i + \alpha_i) - \frac{1}{2} \sum_{i,j=1}^n (\hat{\alpha}_i - \alpha_i) (\hat{\alpha}_j - \alpha_j) (\mathbf{x}_i^T \mathbf{x}_j + \frac{1}{C} \delta_{ij})$$

(1.14)

30

ここで、係数 i を以下のように置き換えると、式 (1.14) は、最終的に以下の式 (1.15) のように書き換えることができる。また、式 (1.11) により、線形回帰を近似する関数 $f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \mathbf{x}$ は、以下の式 (1.16) のようになる。

【 0 0 5 0 】

【 数 7 】

40

係数 $\alpha_i \leftarrow \hat{\alpha}_i - \alpha_i$ の置き換えを行い、さらに、 $\hat{\alpha}_i \alpha_i = 0$ の関係を用いると、

$$\max_{\alpha} \sum_{i=1}^n y_i \alpha_i - \varepsilon \sum_{i=1}^n |\alpha_i| - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j (x_i^T x_j + \frac{1}{C} \delta_{ij}) \quad (1.15)$$

$$f(x, \alpha) = \sum_{i=1}^n \alpha_i x_i^T x \quad (1.16)$$

10

図2は、線形近似関数とサポートベクトルとの関係を示す概念図である。

【0051】

図2において、「 \square 」がサポートベクトルであり、これらの近似関数による誤差の大きさは「 \square 」よりも大きい。「 \square 」は近似関数による誤差の大きさが「 \square 」よりも小さいデータであり、これらは直接近似関数には影響しない。

【0052】

1 - 2 非線形回帰問題の学習

1 - 2 - 1 非線形回帰問題

以上の説明では、サポートベクトル回帰 (SVR) を用いた線形回帰について説明した。以下では、それを応用したSVRを用いた非線形回帰について説明する。この非線形回帰問題は、以下のように定式化できる

20

【問題】

m 入力 1出力の未知非線形回帰 $f: x \rightarrow y$ があり、トレーニングデータ $(x_1, y_1), \dots, (x_n, y_n)$ $x_i \in \mathbb{R}^m, y_i \in \mathbb{R}$ (1.17)

が与えられているとする。このとき、トレーニングデータを用いてこの非線形回帰を近似する関数 $f(x, w) = w^T x$ を求める。ここでは、損失関数として、二次不感損失関数を用いることにする。

【0053】

1 - 2 - 2 特徴空間

一般的に複雑な問題に應用するには、入力空間内で考えるよりも表現能力が優れている空間で考える必要がある。このような手法は高次元化と呼ばれている。

30

【0054】

ここで、非線形回帰問題に対応するために、入力空間から高次元特徴空間への非線形写像を導入する。入力ベクトル x を入力空間 X から特徴空間 F へ写像し、その特徴空間でのベクトル (x) をこれまでの入力ベクトル x とみなすことにより入力空間ではなく特徴空間で線形回帰問題を取り扱う。ただし、 $\phi: X \rightarrow F$ は、入力空間からある特徴空間への非線形写像である。

【0055】

この作業は、実質的には元の入力空間において非線形回帰問題を解いていることになる。

40

【0056】

ヒルベルトシュミット理論によるとヒルベルト空間における内積は次のような等価表現ができる。

【0057】

【数8】

$$(\phi(x_1), \phi(x_2)) = \sum_{i=1}^n a_i \phi_i(x_1) \phi_i(x_2) \leftrightarrow K(x_1, x_2) \quad (1.18)$$

$$a_i \geq 0 \quad (1.19)$$

$$K(x_i, x_j) = \phi(x_i)^T \phi(x_j) \quad (1.20)$$

以下に示すとおり、特徴空間で線形問題を考えると、式(1.21)となり、また、線形回帰を近似する関数 $f(x, w) = w^T \phi(x)$ は、式(1.22)のようになる。この式(1.22)は、入力空間における線形回帰問題となっており、式(1.20)の関係により、最終的に式(1.23)となる。さらに、式(1.23)をベクトル表記すると、式(1.24)となり、この非線形回帰問題を解くと、式(1.25)の非線形近似関数を得ることができる。

【0058】

【数9】

$$\max_{\alpha} \sum_{i=1}^n y_i \alpha_i - \varepsilon \sum_{i=1}^n |\alpha_i| - \frac{1}{2} \sum_{i,j=1}^n (\alpha_i \alpha_j \phi(x_i)^T \phi(x_j) + \frac{1}{C} \delta_{ij}) \quad (1.21)$$

$$f(\phi(x), \alpha) = \sum_{i=1}^n \alpha_i \phi(x_i)^T \phi(x) \quad (1.22)$$

$$\max_{\alpha} \sum_{i=1}^n y_i \alpha_i - \varepsilon \sum_{i=1}^n |\alpha_i| - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j (K(x_i, x_j) + \frac{1}{C} \delta_{ij}) \quad (1.23)$$

$$\max_{\alpha} y^T \alpha - \varepsilon \|\alpha\|_1 - \frac{1}{2} \alpha^T \left(K + \frac{I}{C} \right) \alpha \quad (1.24)$$

$$f(x, \alpha) = \sum_{i=1}^n \alpha_i K(x, x_i) \quad (1.25)$$

ただし、出力 $y = [y_1, \dots, y_n]^T$, 係数 $\alpha = [\alpha_1, \dots, \alpha_n]^T$,

カーネル行列 $K:K$ の i 行 j 列目の要素 $K_{ij} = K(x_i, x_j)$ である。

ここで、注目すべき点は、特徴空間内での内積計算を入力空間内での計算に置き換えることができるカーネル関数 $K(x_i, x_j)$ を用いることにより、双対問題の式(1.21)やそれを解くことによって得られる近似関数には $\phi(x)$ が含まれないので、実際には入力空間から特徴空間への非線形写像を計算する必要はない点である。これは、「カーネルトリック」と呼ばれている。

【0059】

このような特徴があるため、高次元化を導入することにより、サポートベクトル回帰(SVR)では特徴空間の次元を非常に大きくとることもでき、さらにモデルの複雑さと特徴空間の次元とを独立に制御することが可能となる。このことより、SVRはいわゆる次元の呪いから逃れることができる。

【0060】

図3は、非線形近似関数とサポートベクトルを示す概念図である。図3でも、「 α_1 」がサポートベクトルであり、これらの近似関数による誤差の大きさは α_1 よりも大きい。「 α_2 」は近似関数による誤差の大きさが α_1 よりも小さいデータであり、これらは直接近似関数には影響しない。

【0061】

1-2-3 サポートベクトルと係数の関係

二次 不感損失関数を用いたSVR問題では、式(1.14)から式(1.15)に係数の置換えを行う前においては、式(1.11)~(1.13)のKKT条件が成り立っていることから、係

10

20

30

40

50

数の置換え後においては結局以下の関係式が成り立つ。

【 0 0 6 2 】

【数 1 0】

$$\hat{\xi}_i - \xi_i = \frac{\alpha_i}{C} \quad i=1, \dots, n \quad (1.26)$$

また、二次 不感損失関数は、不感帯 を有するので、推定誤差の大きさが よりも小さい、以下の式 (1.27) を満たすデータセット (x_i, y_i) はサポートベクトルから外れる必要がある。このため、得られた解によるサポートベクトルの近似誤差の大きさは、式 (1.28) のようになる。

【 0 0 6 3 】

【数 1 1】

$$\left| y_i - \sum_{j=1}^n \alpha_j K(x_i, x_j) \right| \leq \varepsilon \quad (1.27)$$

$$\left| y_i - \sum \alpha_j K(x_i, x_j) \right| = \left| \hat{\xi}_i - \xi_i \right| + \varepsilon \quad i=1, \dots, n \quad (1.28)$$

さらに、ラグランジェ関数を式 (1.10) のようにおいたので、以下の式 (1.29) および式 (1.30) の符号が等しくなる。このため、式 (1.23) の解であるサポートベクトルの係数 α_i は、式 (1.31) を満たすことになる。

【 0 0 6 4 】

【数 1 2】

$$y_i - \sum_{j=1}^n \alpha_j K(x_i, x_j) \quad (1.29)$$

$$\hat{\xi}_i - \xi_i = \frac{\alpha_i}{C} \quad (1.30)$$

$$y_i - \sum_{j=1}^n \alpha_j K(x_i, x_j) - \varepsilon \cdot \text{sign}(\alpha_i) = \frac{\alpha_i}{C} \quad i = 1, \dots, n \quad (1.31)$$

2 - 1 サポートベクトル回帰の学習に対する本発明の方法

以上の説明の方法をそのまま実行しようとする、S V R 問題においてトレーニングデータが多い場合、問題を解くために非常に多くの時間を費やしてしまう。そこで、本発明においては、S V R 問題を効率良く解くために、以下に説明するとおり、ニュートン・ラフソン (Newton - Raphson) 法を変形させた基本的な更新式を求め、その更新式を用いた逐次形アルゴリズムを採用する。

【 0 0 6 5 】

2 - 1 - 1 問題設定

ここでは、二次 不感損失関数を使用する次の問題を解く方法について説明する。

【 0 0 6 6 】

[問題]

m 入力 1 出力の未知非線形回帰 $f : x \rightarrow y$ があり、トレーニングデータ

$(x_1, y_1), \dots, (x_n, y_n) \quad x_i \in R^m, y_i \in R$

が与えられているとする。このとき、トレーニングデータを用いてこの非線形回帰を近似する式 (2.1) の関数を求めるために、式 (2.2) で表される最大化問題を解く。

10

20

30

40

50

【 0 0 6 7 】

【数 1 3】

$$\text{SVR} : f(x, \alpha) = \sum_{i=1}^n \alpha_i K(x, x_i) \quad (2.1)$$

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^n y_i \alpha_i - \varepsilon \sum_{i=1}^n |\alpha_i| - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j (K(x_i, x_j) + \frac{1}{C} \delta_{ij}) \quad (2.2)$$

10

SVRでは、誤差を評価するとき不感帯を設けているので、この問題を解いて得られた $f(x, \cdot)$ とトレーニングデータには、以下に述べるような関係が成り立っていないなければならない。

【 0 0 6 8 】

すなわち、データセット (x_i, y_i) が不感帯の内側にあるならば、つまり、 $|y_i - f(x_i, \cdot)| < \varepsilon$ を満たすならば、係数は $\alpha_i = 0$ となる。

【 0 0 6 9 】

逆に、データセット (x_i, y_i) が不感帯の外側にあるならば、つまり、 $|y_i - f(x_i, \cdot)| > \varepsilon$ を満たすならば、係数 α_i は、上述した式(1.31)を満たす。

【 0 0 7 0 】

20

以下に述べるサポートベクトル回帰の学習では、上記の関係を考慮して、不必要となったデータセットをサポートベクトルから除外していく方法をとっている。

【 0 0 7 1 】

2 - 1 - 2 バッチ処理

以下では、はじめに n 個のトレーニングデータからいくつかのデータセットを取り出し、その取り出したデータセットだけを用いてSVRを求める方法について説明する。

【 0 0 7 2 】

(ニュートン・ラフソン法の修正)

以下、二次不感損失関数に対する双対問題をニュートン・ラフソン法を変形させて更新式を用いて解く方法について説明する。

30

【 0 0 7 3 】

式(2.2)の最大化問題をベクトル表記すると、式(2.3)のようになり、かつ、評価関数 $W(\cdot)$ をテイラー展開すると、式(2.4)のようになる。

【 0 0 7 4 】

その上で、 $W(\cdot)$ を α^i の近傍で二次関数近似を行ったとして、上式(2.4)の第3項までで打ち切る。

【 0 0 7 5 】

【数 1 4】

$$\max_{\alpha} W(\alpha) = y^T \alpha - \varepsilon \|\alpha\|_1 - \frac{1}{2} \alpha^T \left(K + \frac{I}{C} \right) \alpha \quad (2.3)$$

40

$$W(\alpha) = W(\alpha^i) + \frac{\partial W}{\partial \alpha^T} (\alpha - \alpha^i) + \frac{1}{2} (\alpha - \alpha^i)^T \frac{\partial^2 W}{\partial \alpha \partial \alpha^T} (\alpha - \alpha^i) + \dots \quad (2.4)$$

この二次関数近似の最小値を与える $\alpha = \alpha^i$ は、以下の式(2.5)の条件により与えられ、そのような最小値を与える $\alpha = \alpha^i$ は、逐次的に以下の式(2.6)の関係を満たすことになる(ニュートン・ラフソン法)。

【 0 0 7 6 】

50

【数 1 5】

$$\frac{\partial W(\alpha^{i+1})}{\partial \alpha} = 0 \quad (2.5)$$

$$\frac{\partial W(\alpha^i)}{\partial \alpha} + \frac{\partial^2 W(\alpha^i)}{\partial \alpha \partial \alpha^T} (\alpha^{i+1} - \alpha^i) = 0$$

$$\alpha^{i+1} = \alpha^i - \left[\frac{\partial^2 W(\alpha^i)}{\partial \alpha \partial \alpha^T} \right]^{-1} \frac{\partial W(\alpha^i)}{\partial \alpha} \quad (2.6)$$

ただし、

$$\frac{\partial W}{\partial \alpha} = y - \left(K + \frac{I}{C} \right) \alpha - \varepsilon \cdot \text{sign}(\alpha) \quad (2.7)$$

$$\frac{\partial^2 W}{\partial \alpha \partial \alpha^T} = -K - \frac{I}{C} - \varepsilon \cdot \text{diag}(\text{sign}'(\alpha)) \quad (2.8)$$

$$\text{sign}(\alpha) = [\text{sign}(\alpha_1), \dots, \text{sign}(\alpha_n)]^T, \text{sign}'(\alpha) = [\text{sign}'(\alpha_1), \dots, \text{sign}'(\alpha_n)]^T$$

$$\text{sign}(x) \approx \frac{2}{1 + \exp(-ax)} - 1, \quad a \gg 1 \quad (2.9)$$

$$\text{sign}'(x) = \frac{\partial \text{sign}(x)}{\partial x} \quad (2.10)$$

$$\approx 2a \left(\frac{1}{1 + \exp(-ax)} \right) \left(1 - \frac{1}{1 + \exp(-ax)} \right) \quad (2.11)$$

(の更新方法 (初期値 0 の場合))

ニュートン・ラフソン法に基づき を求める際、初期値 $\alpha^0 = 0$ の場合について考える。式 (2.6) で、 $\alpha^0 = 0$ とすると、以下の式 (2.12) および式 (2.13) を満たすことになるが、本発明においては、 $\alpha^0 = 0$ とおいて、式 (2.14) のように修正する。

【 0 0 7 7 】

【数 1 6】

$$\alpha^1 = \alpha^0 + \left(K + \frac{I}{C} + \varepsilon \cdot \text{diag}(\text{sign}'(\alpha^0)) \right)^{-1} \left(y - \left(K + \frac{I}{C} \right) \alpha^0 - \varepsilon \cdot \text{sign}(\alpha^0) \right) \quad (2.12)$$

$$= \left(K + \frac{I}{C} + \varepsilon \cdot \text{diag}(\text{sign}'(\alpha^0)) \right)^{-1} y \quad (2.13)$$

$$\alpha^1 = \left(K + \frac{I}{C} \right)^{-1} y \quad (2.14)$$

その後、さらに式 (2.6) を用いて更新すると、以下の式 (2.15) および (2.16) のようになる。さらに、 $\alpha^i = 0$ のとき、式 (2.17) が成り立つので、式 (2.16) は、式 (2.18) および (2.19) で表されるようになる。

【 0 0 7 8 】

【数 1 7】

10

20

30

40

$$\alpha^2 = \alpha^1 + \left(\mathbf{K} + \frac{\mathbf{I}}{\mathbf{C}} + \varepsilon \cdot \text{diag}(\text{sign}'(\alpha^1)) \right)^{-1} \left(\mathbf{y} - \left(\mathbf{K} + \frac{\mathbf{I}}{\mathbf{C}} \right) \alpha^1 - \varepsilon \cdot \text{sign}(\alpha^1) \right) \quad (2.15)$$

$$= \alpha^1 - \varepsilon \left(\mathbf{K} + \frac{\mathbf{I}}{\mathbf{C}} + \varepsilon \cdot \text{diag}(\text{sign}'(\alpha^1)) \right)^{-1} \text{sign}(\alpha^1) \quad (2.16)$$

$$\text{sign}'(\alpha_i^1) = 0 \quad (2.17) \quad 10$$

$$\alpha^2 = \alpha^1 - \varepsilon \left(\mathbf{K} + \frac{\mathbf{I}}{\mathbf{C}} \right)^{-1} \text{sign}(\alpha^1) \quad (2.18)$$

$$= \left(\mathbf{K} + \frac{\mathbf{I}}{\mathbf{C}} \right)^{-1} \left(\mathbf{y} - \varepsilon \cdot \text{sign}(\alpha^1) \right) \quad (2.19)$$

このように求めた係数ベクトル で、以下の式 (2.20) が成り立っている場合、式 (2.21) のようになるので、上述した式 (1.31) を満たしていることがわかる。

【 0 0 7 9 】

【 数 1 8 】

20

$$\text{sign}(\alpha^1) = \text{sign}(\alpha^2) \quad (2.20)$$

$$\alpha^2 = \left(\mathbf{K} + \frac{\mathbf{I}}{\mathbf{C}} \right)^{-1} \left(\mathbf{y} - \varepsilon \cdot \text{sign}(\alpha^2) \right) \quad (2.21)$$

しかし、求めた解が式 (2.20) を満たしていない場合、その解は式 (1.31) を満たしてなく、解としてはふさわしくない。

【 0 0 8 0 】

(余分なデータセットの除去方法)

30

次に、式 (2.20) を満たしていない場合について考えることにする。

【 0 0 8 1 】

式 (2.20) を満たしていない場合は、求めた解は式 (1.31) を満たしてなく、KKT条件を満たさないので解ではないことになる。そこで、このような場合は、余分なデータセット (x_i, y_i) が現在のサポートベクトルの候補の中に存在していると考え、現在のサポートベクトルの候補の中から余分なデータセットを取り除くことにする。

【 0 0 8 2 】

以下では、さらに、余分なデータセットの選出方法について以下に述べる。

まずはじめに、先ほどの更新式を用いて、以下の式 (2.22) に基づいて、係数ベクトル n_{sv} を求める (初期値 : 0)。

40

【 0 0 8 3 】

次に、係数ベクトル n_{sv} の各要素はゼロではない ($n_i > 0, i = 1, \dots, n_{sv}$) と考え、さらに更新式 (2.6) にステップサイズ ($0 < \eta < 1$) を導入して、以下の式 (2.23) のような新たな更新式をつくる。

【 0 0 8 4 】

【 数 1 9 】

$$\alpha_{n_{sv}} = \left(\mathbf{K}_{n_{sv}} + \frac{\mathbf{I}_{n_{sv}}}{\mathbf{C}} \right)^{-1} \mathbf{y}_{n_{sv}} \quad (2.22)$$

$$\alpha_{n_{sv}} \leftarrow \alpha_{n_{sv}} + \eta \left(\mathbf{K}_{n_{sv}} + \frac{\mathbf{I}_{n_{sv}}}{\mathbf{C}} \right)^{-1} \left(\mathbf{y}_{n_{sv}} - \left(\mathbf{K}_{n_{sv}} + \frac{\mathbf{I}_{n_{sv}}}{\mathbf{C}} \right) \alpha_{n_{sv}} - \varepsilon \cdot \text{sign}(\alpha_{n_{sv}}) \right) \quad (2.23)$$

この更新式を用いて係数ベクトル \mathbf{n}_{sv} を更新する際、更新前後で係数ベクトル \mathbf{n}_{sv} の各要素の符号が変化しない範囲で更新を行うことにする。 10

【0085】

次に、ステップサイズの決め方について説明する。

ステップサイズは、1に近ければ近いほど評価関数が大きな値となるので、できる限り1に近いことが望ましい。そこで、次のようにしてそのステップサイズを求める。

【0086】

まず、以下の式(2.24)により、ステップサイズベクトルを計算する。

【0087】

【数20】

$$\eta = -\alpha_{n_{sv}} ./ \left(\mathbf{K}_{n_{sv}} + \frac{\mathbf{I}_{n_{sv}}}{\mathbf{C}} \right)^{-1} \left(\mathbf{y}_{n_{sv}} - \left(\mathbf{K}_{n_{sv}} + \frac{\mathbf{I}_{n_{sv}}}{\mathbf{C}} \right) \alpha_{n_{sv}} - \varepsilon \cdot \text{sign}(\alpha_{n_{sv}}) \right) \quad (2.24)$$

ただし、記号 ./ は、例えば $A ./ B$ の演算の場合に、行列 A の

各要素を対応する行列 B の各要素で割ることを意味する。

ただし、ステップサイズベクトル $\boldsymbol{\eta} = [\eta_1, \dots, \eta_{n_{sv}}]^T$ であり、 η_i は、更新式(2.23)においてステップサイズ η_i で、係数ベクトル \mathbf{n}_{sv} を更新すると $\eta_i = 0$ となる値である。 30

【0088】

つぎに、ステップサイズベクトル $\boldsymbol{\eta}$ の要素の中で、0以上で、かつ、最も小さい要素 η_i をステップサイズとして更新を行う。このとき、 $\eta_i = 0$ であり、データセット (x_i, y_i) を余分なデータセットであるとし、以下の式(2.25)のようにサポートベクトルから外すことにする。

【0089】

【数21】

$$\boldsymbol{\alpha} \leftarrow [\alpha_1, \dots, \alpha_{i-1}, \alpha_{i+1}, \dots, \alpha_{n_{sv}}] \quad (2.25)$$

40

これに伴いサポートベクトルの個数が減少することになる ($n_{sv} \rightarrow n_{sv} - 1$)。

【0090】

サポートベクトルの個数が減少した後、再び更新式(2.23)より係数ベクトル \mathbf{n}_{sv} を更新する。このときも先ほどと同じように、更新前後で係数ベクトル \mathbf{n}_{sv} の各要素の符号が変化しない範囲で更新を行うために式(2.24)からステップサイズ η を求める。

【0091】

ステップサイズベクトル $\boldsymbol{\eta}$ の要素の中で0以上で最も小さい要素 η_i が1以上の場合、更新式(2.23)において $\eta_i = 1$ で更新した係数ベクトル \mathbf{n}_{sv} は更新前後で各要素の符号が変化しないことがわかる。

50

【 0 0 9 2 】

このとき、係数ベクトル n_{sv} が決定する。逆にステップサイズベクトル の要素の中で 0 以上で最も小さい要素 i が 1 よりも小さい場合、先ほどと同じく更新式 (2.23) においてステップサイズ $= i$ で係数ベクトル n_{sv} を更新し、データセット (x_i, y_i) をサポートベクトルから外すことにする。以降、係数ベクトル n_{sv} が決定するまでこの作業を繰り返し行う。

【 0 0 9 3 】

2 - 1 - 3 逐次処理

以上の説明では、 n 個のトレーニングデータからいくつかのデータセットを取り出し、その取り出したデータセットだけから SVR を求める方法について説明した。しかし、バッチ処理では逆行列の計算を行っているため、扱うデータ数には限界がある。そこで、このような問題を回避するために、次に逐次処理について説明する。

【 0 0 9 4 】

ここでは、既に以下の式 (2.26) のような SVR による近似関数が存在しており、新たなトレーニングデータ (x_s, y_s) を用いて、この SVR による近似関数を更新する方法について説明する。

【 0 0 9 5 】

【数 2 2】

$$\text{SVR} : f(x, \alpha) = \sum_{i=1}^{n_{sv}} \alpha_i K(x, x_i) \quad (2.26)$$

まず、はじめに、新たなトレーニングデータ (x_s, y_s) が不感帯の内側にあるかどうかを調べる。

【 0 0 9 6 】

もし、以下の式 (2.27) が成り立つならば、不感帯の内側にあることになり SVR の更新は必要でないが、新たなトレーニングデータ (x_s, y_s) が不感帯の外側にある場合、つまり以下の式 (2.28) が成り立つ場合、SVR の更新が必要となる。

【 0 0 9 7 】

【数 2 3】

$$\left| y_s - \sum_{i=1}^{n_{sv}} \alpha_i K(x_s, x_i) \right| \leq \varepsilon \quad (2.27)$$

$$\left| y_s - \sum_{i=1}^{n_{sv}} \alpha_i K(x_s, x_i) \right| > \varepsilon \quad (2.28)$$

そこで、新たなトレーニングデータ (x_s, y_s) が不感帯の外側にある場合について考えることにする。

【 0 0 9 8 】

このとき、 $(x_1, y_1), \dots, (x_{n_{sv}}, y_{n_{sv}}), (x_s, y_s)$ をサポートベクトルであるとみなす。

【 0 0 9 9 】

先ほど示した式 (2.23) を用いて、係数ベクトル を更新しようとした場合、新たなトレーニングデータ (x_s, y_s) の係数 s が決まっていなかったのでこの値の初期値を 0 とする。

【 0 1 0 0 】

しかし、このままでは、式 (2.23) を計算する際、 $\text{sign}(s) = 0$ となり、このまま係数ベクトル を更新して求めても、KKT 条件を満たさない。

【 0 1 0 1 】

そこで、更新式 (2.23) を次のように変更する。

【 0 1 0 2 】

【 数 2 4 】

$$\alpha_{n_{sv}} \leftarrow \alpha_{n_{sv}} + \eta \left(K_{n_{sv}} + \frac{I_{n_{sv}}}{C} \right)^{-1} \left(y_{n_{sv}} - \left(K_{n_{sv}} + \frac{I_{n_{sv}}}{C} \right) \alpha_{n_{sv}} - \varepsilon \cdot \text{Sign} \right) \quad (2.29)$$

ただし、

$$\alpha \leftarrow [\alpha_1 \dots \alpha_{n_{sv}} \ 0]^T \quad (2.30)$$

$$\text{Sign} \leftarrow \left[\text{sign}([\alpha_1 \dots \alpha_{n_{sv}}]) \ \text{sign} \left(y_s - \sum_{i=1}^{n_{sv}} \alpha_i K(x_s, x_i) \right) \right]^T \quad (2.31)$$

この後は、更新式 (2.29) を用いて を更新することを考える。ここでも、更新前後で、係数ベクトル n_{sv} の各要素の符号が変化しない範囲で更新を行うために式 (2.24) を次のように変更し、ステップサイズ を求める。

【 0 1 0 3 】

【 数 2 5 】

$$\eta = -\alpha_{n_{sv}} \cdot \left(K_{n_{sv}} + \frac{I_{n_{sv}}}{C} \right)^{-1} \left(y_{n_{sv}} - \left(K_{n_{sv}} + \frac{I_{n_{sv}}}{C} \right) \alpha_{n_{sv}} - \varepsilon \cdot \text{Sign} \right) \quad (2.32)$$

これ以降は、バッチ処理のところで行ったように の更新を行い、 を決定する。

【 0 1 0 4 】

実際には、本発明においては、以下の前置型非線形歪補償システムの処理において説明するとおり、大量のトレーニングデータセットが与えられている場合についての SVR を求めるアルゴリズムを用いる。

【 0 1 0 5 】

このアルゴリズムは、バッチ処理・逐次処理を一つにまとめた計算方法である。

【 0 1 0 6 】

[非線型デバイス (NLD) の飽和特性を特に考慮しない構成]

図 4 は、本発明の前置型非線形歪補償システムの構成を説明するにあたり、その理解の前提として、非線型デバイス (NLD) の飽和特性を特に考慮しない構成を有する前置型非線形歪補償システム 1000 を示す概略ブロック図である。

【 0 1 0 7 】

図 4 に示した例では、増幅器 16 の出力の歪を前置型非線形歪補償器により補償する構成となっている。

【 0 1 0 8 】

図 4 を参照して、前置型非線形歪補償システム 1000 は、無線周波数信号 S_{IN} を中間周波数信号に変換するダウンコンバータ 2 と、ダウンコンバータ 2 に接続され、ダウンコンバータ 2 の出力する中間周波数信号をサンプリングしてデジタル信号に変換する A/D 変換器 4 と、A/D 変換器 4 に接続され、サンプリングされた中間周波数信号をベースバンド信号 $x(t)$ に変換する直交検波器 6 と、直交検波器 6 に接続され、信号 $x(t)$ を、後述する増幅器 16 の逆特性を用いて、ベースバンド信号 $y(t)$ の実数部 $\text{Re}\{y(t)$

10

20

30

40

50

) } に変換する SVM54 と、直交検波器 6 に接続され、信号 $x(t)$ を、増幅器 16 の逆特性を用いて、ベースバンド信号 $y(t)$ の虚数部 $Im\{y(t)\}$ に変換する SVM58 と、SVM54 および 58 に接続され、変換後のベースバンド信号 $y(t)$ を中間周波数信号に変換する直交変調器 10 と、直交変調器 10 に接続され、直交変調器 10 の出力する中間周波数信号をアナログ信号に変換する D/A 変換器 12 とを含む。

【0109】

なお、本願発明は、図 4 において、ダウンコンバータ 2、A/D 変換器 4 および直交検波器 6 からなる受信部 1 の代わりに、無線送信機のベースバンド信号の出力部が接続されるような態様にも適用可能である。

【0110】

前置型非線形歪補償システム 1000 は、さらに、D/A 変換器 12 に接続され、アナログの中間周波数信号を無線周波数信号 S_0 に変換し、増幅器 16 に供給するアップコンバータ 14 と、増幅器 16 に接続され、増幅器 16 より出力される出力信号 S_{OUT} を中間周波数信号に変換するダウンコンバータ 18 と、ダウンコンバータ 18 に接続され、中間周波数信号をデジタル信号に変換する A/D 変換器 20 と、A/D 変換器 20 に接続され、デジタル信号をベースバンド信号 $z(t)$ に変換する直交検波器 22 とを含む。

【0111】

前置型非線形歪補償システム 1000 は、さらに、直交検波器 22 の出力する信号 $z(t)$ と学習に用いたベースバンド信号ベクトル $V_{z}(p_i)$ ($i = 1, \dots, n_R$) とに基づいて、SVM54 より出力されるベースバンド信号の実数部の推定値 $Re\{y(t)\}$ を出力する SVM26 と、直交検波器 22 の出力する信号 $z(t)$ とベースバンド信号ベクトル $V_{z}(q_j)$ ($j = 1, \dots, n_I$) とに基づいて、SVM58 より出力されるベースバンド信号の虚数部の推定値 $Im\{y(t)\}$ を出力する SVM62 と、SVM54 および SVM26 に接続され、ベースバンド信号の実数部の推定値 $Re\{y(t)\}$ とベースバンド信号 $y(t)$ の実数部 $Re\{y(t)\}$ との差分を計算する減算器 30 と、SVM58 および SVM62 に接続され、ベースバンド信号の虚数部の推定値 $Im\{y(t)\}$ とベースバンド信号 $y(t)$ の虚数部 $Im\{y(t)\}$ との差分を計算する減算器 64 と、直交検波部 6 および直交検波部 22 に接続され、信号ベクトルおよびサポートベクトルのサイズ m と増幅器 16 による遅延時間 d とを求める $m \cdot d$ 決定部 44 と、減算器 30、SVM54、SVM26、直交検波部 22 および $m \cdot d$ 決定部 44 に接続され、減算器 30 の出力、直交検波部 22 の出力および $m \cdot d$ 決定部 44 の出力に基づいて、SVM54 および SVM26 における、増幅器 16 の逆特性を有する関数の係数 $r(i)$ を更新する係数更新部 42 と、減算器 64、SVM58、SVM62、直交検波部 22 の出力および $m \cdot d$ 決定部 44 に接続され、減算器 64 の出力、直交検波部 22 の出力および $m \cdot d$ 決定部 44 の出力に基づいて、SVM58 および SVM62 における、増幅器 16 の逆特性を有する関数の係数 $i(j)$ を更新する係数更新部 66 とを含む。

【0112】

SVM26 は、後に説明する方法に従い、ベースバンド信号の実数部の推定値 $Re\{y(t)\}$ を求める。SVM62 は、SVM26 と同様の方法に従い、ベースバンド信号の虚数部の推定値 $Im\{y(t)\}$ を求める。

【0113】

係数更新部 42 は、 $Re\{y(t)\}$ と $Re\{y(t)\}$ との誤差 $e_R(t)$ が最小になるように係数 $r(i)$ を制御する。係数更新部 66 は、係数更新部 42 と同様に、 $Im\{y(t)\}$ と $Im\{y(t)\}$ との誤差 $e_I(t)$ が最小になるように $i(j)$ を制御する。

【0114】

なお、直交検波器 6 をアナログで実現し、ダウンコンバータ 2 の出力を直交検波器 6 で受け、直交検波器 6 の出力を A/D 変換器 4 でデジタルに変換し、その出力を SVM8 に供給する構成であっても構わない。また、直交変調器 10 をアナログで実現し、SVM54 および SVM58 の出力を D/A 変換器 12 が受け、D/A 変換器 12 の出力を直交変調

10

20

30

40

50

器 10 で受け、直交変調器 10 の出力をアップコンバータ 12 に供給する構成であっても構わない。さらに、直交検波器 22 をアナログで実現し、ダウンコンバータ 18 の出力を直交検波器 22 で受け、直交検波器 22 の出力を A/D 変換器 20 で受け、A/D 変換器 20 の出力を SVM 26 および SVM 62 に供給する構成であっても構わない。

【 0 1 1 5 】

(前置型非線形歪補償システム 1000 の動作)

以下、前置型非線形歪補償システム 1000 の動作について説明する。

【 0 1 1 6 】

まず、図 4 において、信号 S_{IN} 、 S_0 および S_{OUT} は、式 (1) ~ (3) でそれぞれ表わされる。

【 0 1 1 7 】

$$S_{IN} = \text{Re} \{ x(t) \exp(j 2\pi (f_{RF} t)) \}$$

: 前置型非線形歪補償器への入力信号 ... (1)

$$S_0 = \text{Re} \{ y(t) \exp(j 2\pi (f_{RF} t)) \}$$

: 歪補償された信号 ... (2)

$$S_{OUT} = \text{Re} \{ z(t) \exp(j 2\pi (f_{RF} t)) \}$$

: 増幅器 16 からの出力信号 ... (3)

$x(t)$ 、 $y(t)$ および $z(t)$ は、それぞれ帯域信号 S_{IN} 、 S_0 および S_{OUT} の複素包絡線 (ベースバンド信号) である。j は虚数を表わす。 f_{RF} は無線周波数 (Radio Frequency) を表わす。t は時刻を表わす。

【 0 1 1 8 】

[非線形歪補償器の動作原理]

$P = |y(t)|^2 / 2$ として、増幅器 16 のベースバンド領域における入出力特性を f を複素関数として $f(P)$ と置くと、ベースバンド信号 $y(t)$ および $z(t)$ の関係は、下記の式 (4) のように表すことができる。

【 0 1 1 9 】

さらに、一般に、非線形歪補償器では、 $f(P)$ に対し、次式 (5) の関係を満たす入力特性 $g(P)$ を求めることを目的とする。仮に、入力特性 $g(P)$ を得ることができれば、次式 (6) および (7) が成立し、 S_{IN} に対して S_{OUT} を線形化することができる。

【 0 1 2 0 】

【 数 2 6 】

$$z(t) = f(P)y(t) \tag{4}$$

: ただし、 $P = \frac{1}{2}|y(t)|^2$ 、 $f(P)$ は複素関数。

$f(P) \cdot g(P) = k$ k は増幅器 16 の利得 (実数) を表わす。

(5)

$$y(t) = g(P)x(t) \tag{6}$$

$$z(t) = f(P)g(P)x(t) = kx(t) \tag{7}$$

前置型非線形歪補償システム 1000 では、 $g(P)$ を、SVM 26 および係数更新部 42 ならびに SVM 62 および係数更新部 66 にて推定する。具体的には、 $y(t)$ と $z(t)$ より推定した $y'(t)$ との誤差 $e(t)$ が、実部および虚部のそれぞれにおいて最小になるように、SVM 26 および SVM 62 の関数の係数を更新する。この計算が収束すれば、SVM 26 および SVM 62 により近似される特性は、増幅器 16 の逆特性に相当することとなる。

【 0 1 2 1 】

10

20

30

40

50

[逆特性関数の計算]

上述したような逆特性関数の計算のための条件を以下に整理しておく。

・非線形の歪補償のための計算は時間連続信号ではなく離散信号に対して行なわれる。ハードウェア化の際は、A / D変換器を使用し、連続信号を離散信号に変換し、計算後、離散信号を連続信号にD / A変換器により復元する。

・信号処理は帯域系ではなくベースバンド系または中間周波数信号にて行なわれる。ベースバンド系の場合には、増幅器 16 は等価的に複素ベースバンド信号 $y(t)$ を受け、複素信号 $z(t)$ を出力する系と見なすことができる。

【 0 1 2 2 】

[サポートベクトルマシンの動作説明]

時刻 kT (T : A / D変換およびD / A変換のサンプリング間隔) のとき、サポートベクトルマシン SVM54 の出力 $\text{Re}\{y(kT)\}$ および SVM58 の出力 $\text{Im}\{y(kT)\}$ 、ならびにサポートベクトルマシン SVM26 の出力 $\text{Re}\{y_s(kT)\}$ および SVM62 の出力 $\text{Im}\{y_s(kT)\}$ は、次式 (8) ~ (11) により得られる。

【 0 1 2 3 】

【 数 2 7 】

SVM54

$$\text{Re}\{y(kT)\} = \sum_{i=1}^{n_R} \alpha_R(i) K\{V_z(p_i T), V_x(kT)\} \quad (8)$$

10

20

SVM58

$$\text{Im}\{y(kT)\} = \sum_{j=1}^{n_I} \alpha_I(j) K\{V_z(q_j T), V_x(kT)\} \quad (9)$$

SVM26

$$\text{Re}\{y_s(kT)\} = \sum_{i=1}^{n_R} \alpha_R(i) K\{V_z(p_i T), V_z(kT)\} \quad (10)$$

SVM62

$$\text{Im}\{y_s(kT)\} = \sum_{j=1}^{n_I} \alpha_I(j) K\{V_z(q_j T), V_z(kT)\} \quad (11)$$

30

ただし、

$$V_z(p_i T) = [z(p_i T), z((p_i - 1)T), \dots, z((p_i - m + 1)T)]^T \quad (12)$$

$(i = 1, \dots, n)$

$$V_z(kT) = [z(kT), z((k - 1)T), \dots, z((k - m + 1)T)]^T \quad (13)$$

$$y_s(kT) = y((k - d)T) \quad d: \text{増幅器 16 の入出力間の遅延} \quad (14)$$

ここで、変数の接頭辞 " V_z " は、その変数がベクトルであることを表わす。また、 m は信号ベクトルおよび後述するサポートベクトルのサイズ、 n_R 、 n_I は、それぞれ実部および虚部のサポートベクトルマシンのサポートベクトルの数であり、 p_i は i 番目のサポートベクトルを取得した時間であり、 q_j は j 番目のサポートベクトルを取得した時間である。

40

【 0 1 2 4 】

$V_z(p_i T)$ ($i = 1, \dots, n_R$)、 $V_z(q_j T)$ ($j = 1, \dots, n_I$)

は増幅器 16 の入出力特性を学習した結果、時刻 kT までに抽出されたサポートベクトルである。

【 0 1 2 5 】

$\alpha_R(i)$ 、 $\alpha_I(j)$ は、それぞれ $V_z(p_i T)$ 、 $V_z(q_j T)$ に対応した重み係

50

数（実数）である。一般に、 $w_R(i)$ （または、 $w_I(j)$ ）が零の時、対応する $V_z(p_i^T)$ （または、 $V_z(q_j^T)$ ）は計算に加える必要がなく廃却しても構わない。また、非零の係数 $w_R(i)$ （または、 $w_I(j)$ ）の個数をサポートベクトルマシンの次元と呼ぶ。例えば、非零の係数の個数が73であればサポートベクトルマシンの次元数は73である。さらに、この次元数は学習データを増やしても一定値以上に増えることがない。この上界値のことをVC（Vapnik and Chervonenkis）次元と呼ぶ。

【0126】

$K(V_x, V_y)$ は「内積カーネル」の関数であり、前置型非線形歪補償システム1000では、内積カーネルの線形和である式(8)~(11)を、適切な係数 $w_R(i)$ 、 $w_I(j)$ を選択することにより、増幅器16の入出力特性の逆特性 $g(P)$ に適合させることを目標とする。このように未知の関数にシステムの入出力関数を適合させるような問題全般を総称して「回帰問題」という。次式(15)~(17)は、一般的に用いられる内積カーネルの例である。

10

【0127】

【数28】

$$K(V_x, V_y) = \exp(-\|V_x - V_y\|^2 / 2\sigma^2) \quad : \text{Gaussian} \quad (15)$$

$$K(V_x, V_y) = (aV_x^T V_y + 1)^P \quad : \text{Polynomial} \quad (16)$$

$$K(V_x, V_y) = \tanh(aV_x^T V_y - b) \quad : \text{Sigmoid} \quad (17)$$

20

どの内積カーネルが逆関数を表現するのに最も適しているかはシステムにより異なり、実際には動作前に何度か試験をした上で最も特性のよいものを選択する。もちろん、上記以外の関数でも逆関数 $g(P)$ をよりよく推定でき、かつ期待する効果が得られるのであれば、その関数を使用してもよい。

【0128】

[係数更新部42の動作説明]

以下では、サポートベクトルに対応する係数 $w_R(i)$ 、 $w_I(j)$ の更新計算を行う係数更新部42および係数更新部66の動作について説明する。

30

【0129】

上述したとおり、係数更新部42は、 $\text{Re}\{y(t)\}$ と $\text{Re}\{y^*(t)\}$ との誤差 $e_R(t)$ が最小になるように係数 $w_R(i)$ を制御する。一方、係数更新部66は、 $\text{Im}\{y(t)\}$ と $\text{Im}\{y^*(t)\}$ との誤差 $e_I(t)$ が最小になるように係数 $w_I(j)$ を制御する。

【0130】

【数29】

SVM54, SVM26

$$e_R(p_n) = \text{Re}\{y(p_n)\} - \text{Re}\{ys'(p_n)\} \quad (18)$$

40

(ここで、 $ys(p_n), ys'(p_n)$ は、 $ys(p_n^T), ys'(p_n^T)$ で、

$T=1$ とした例の表現。以下同じ)

SVM58, SVM62

$$e_I(q_n) = \text{Re}\{y(q_n)\} - \text{Re}\{ys'(q_n)\} \quad (19)$$

なお、以下では、表記を簡単にするため $T=1$ としている。このようにしても一般性を失うことはない。

50

【 0 1 3 1 】

係数 $\alpha_R(i)$ は、次式 (20) で表わされる評価関数 W を最大化するという最適化問題を解くことにより得られる。また、係数 $\alpha_I(j)$ についても同様である。

【 0 1 3 2 】

【 数 3 0 】

評価関数 W

$$W = \sum_{i=1}^{n_R} \alpha_R(i) \operatorname{Re}\{y_s(p_i)\} - \varepsilon \sum_{i=1}^{n_R} |\alpha_R(i)| - \frac{1}{2} \sum_{i,j=1}^{n_R} \alpha_R(i) \alpha_R(j) \{K(\operatorname{Re}(V_{-z}(p_i T)), \operatorname{Re}(V_{-z}(p_j T))) + \frac{\delta_{ij}}{C}\} \quad (20)$$

$$\text{ただし、} \delta_{ij} = \begin{cases} 1 & (i=j) \\ 0 & (i \neq j) \end{cases}$$

ε : 雑音に対するマージン (定数)

C : モデルの複雑さと制約条件の緩和 (トレーニング誤差)

との間のトレードオフを設定する正則化パラメータ

$$y_s(p_i) = y(p_i - d)$$

ただし、定数 C と定数 ε は、通常、実験的に求められる。

【 0 1 3 3 】

増幅器 16 の入出力特性の学習が完了するまでの間はサポートベクトルの追加・削除が繰返されるが、これと対応して係数 $\alpha_R(i)$, $\alpha_I(j)$ の値を更新する必要がある。

【 0 1 3 4 】

以下では、係数更新部 42 の動作を例にとって説明する。係数更新部 66 の動作も、入力される変数が異なるのみで、基本的に係数更新部 42 と同様である。

【 0 1 3 5 】

なお、後に詳しく説明するように、係数更新部 42 の更新の計算の特徴の 1 つは、以下のようにして逐次的に係数 $\alpha_R(i)$ の値を更新することである。

(1) 更新の計算の途中において、係数更新部 42 は、 $n - 1$ 組のサポートベクトル $V_{-z}(p_i)$ および係数 $\alpha_R(i)$ を取得している状態となる。

(2) 係数更新部 42 は、さらに、 n 個目のサポートベクトルの候補としてデータ $V_{-z}(p_n)$ を入手する。

(3) $V_{-z}(p_n)$ をサポートベクトルと見なし、その係数を $\alpha_R(n) = 0$ とおく。

(4) $V_{-z}(p_n)$ が加わったことによる最適解からのずれ (摂動) を補正するため、 $\alpha_R(i)$ ($i = 1, \dots, n$) に関して誤差分 (摂動分) $\Delta \alpha_R(i)$ を計算する。

【 0 1 3 6 】

上述のように逐次サポートベクトルを抽出することにより、1 回の係数更新において計算対象となる係数の数を削減できるので、計算量を削減することができる。

【 0 1 3 7 】

[係数の更新の具体的な計算方法]

係数の更新方法の具体例を以下に示す。ただし、この具体例は実現の一例であって、必ずしもこの通りである必要はない。

【 0 1 3 8 】

10

20

30

40

50

図5および図6は、このような係数の更新の具体的な計算方法を説明するためのフローチャートである。

【0139】

図5を参照して、係数更新部42は、まず、取得済みのN個の学習データセットからn個のデータセット($V_{-z}(P_i), \text{Re}\{ys(P_i)\}$) ($i = 1 \sim n$)を取り出し、これらをサポートベクトルに採用する(ステップ100)。

【0140】

次に、係数更新部42は、上記サポートベクトルに対応する係数の第1次近似と、その符号ベクトル S_R を、以下の式(21)および(22)により計算する(ステップ102)

$$\begin{aligned} \hat{\alpha}_R &= [\alpha_R(1), \alpha_R(2), \dots, \alpha_R(n)]^T \\ &= \text{inv}\left(\hat{K}_R + \frac{\hat{I}(n)}{C}\right) \cdot \hat{y}_R \end{aligned} \quad (21)$$

$$\hat{S}_R = [\text{sign}(\alpha_R(1)), \dots, \text{sign}(\alpha_R(n))]^T \quad (22)$$

ただし、

$[\dots]^T$: 行ベクトルを転置して列ベクトルに変換する操作

$\text{inv}[\dots]$: 行列 $[\dots]$ に対する逆行列

\cdot : 行列の積

$$\hat{K}_R = \begin{pmatrix} K_R(1,1) & K_R(1,2) & \dots & K_R(1,n) \\ K_R(2,1) & K_R(2,2) & \dots & K_R(2,n) \\ \vdots & & & \vdots \\ K_R(n,1) & K_R(n,2) & \dots & K_R(n,n) \end{pmatrix}$$

$$K_R(i,j) = K\{V_{-z}(p_i), V_{-z}(p_j)\}$$

$\hat{I}(n)$: n行×n列の単位行列

$$\hat{y}_R = [\text{Re}\{ys(p_1), ys(p_2), \dots, ys(p_n)\}]^T \quad (23)$$

さらに、ステップサイズベクトル $[\alpha_R(1), \dots, \alpha_R(n)]^T$ を以下の式(24)および(25)により計算する(ステップS104)。

【0142】

【数32】

10

20

30

40

$$\hat{\eta}_R = [\eta_R(1), \eta_R(2), \dots, \eta_R(n)]^T \quad (24)$$

ただし、 $\eta_R(i) = -\alpha_R(i) / \Delta \alpha_R(i)$

$$\begin{aligned} & [\Delta \alpha_R(1), \Delta \alpha_R(2), \dots, \Delta \alpha_R(n)]^T \\ &= \text{inv} \left(\hat{K}_R + \frac{\hat{I}(n)}{C} \right) \cdot (\hat{y}_R - \epsilon \hat{S}_R) - \hat{\alpha}_R \end{aligned} \quad (25) \quad 10$$

ステップサイズベクトルの要素の中で0以上でもっとも小さい要素を $R(i)$ とし(ステップS106)、要素 $R(i)$ が1以上であるか否かを判定する(ステップS108)。

【0143】

まず、要素 $R(i)$ が1未満である場合、続いて、要素 $R(i)$ を用いて、以下の式(26)により、係数ベクトル $[\alpha_R(1), \dots, \alpha_R(n)]^T$ を更新する(ステップS110)。

【0144】

【数33】

20

$$\hat{\alpha}_R \leftarrow \hat{\alpha}_R + \eta_R(i) \left\{ \text{inv} \left(\hat{K}_R + \frac{\hat{I}(n)}{C} \right) \cdot (\hat{y}_R - \epsilon \hat{S}_R) - \hat{\alpha}_R \right\} \quad (26)$$

その上で、 $(V_z(P), \text{Re}\{y_s(P)\})$ を余分なデータセットとして、サポートベクトルから削除し、係数 $\alpha_R(P)$ も削除する(ステップS112)。このような削除に対応して、 $i >$ について、以下の表にしたがって、 i 番目の学習データを取得した時間、および係数 $\alpha_R(i)$ の番号を読替える。

30

【0145】

【表1】

読替え前	読替え後
P_i	P_{i-1}
$\alpha_R(i)$	$\alpha_R(i-1)$

さらに、 $(n-1)$ を新たに n と読替える処理を行う(ステップS114)。以上の読替えの手続きの後、処理はステップS104に復帰する。

40

【0146】

一方、ステップS108において、要素 $R(i)$ が1以上である場合、以下の式(27)により、改めて係数ベクトルを計算する(ステップS116)。

【0147】

【数34】

$$\hat{\alpha}_R = \text{inv} \left(\hat{K}_R + \frac{\hat{I}(n)}{C} \right) \cdot (\hat{y}_R - \epsilon \hat{S}_R) \quad (27)$$

次に、図6を参照して、新しい学習データセット ($V_{-z}(k), \text{Re}\{y_s(k)\}$) の取得が行われる (ステップS118)。

【0148】

さらに、以下の式(28)により、誤差 e_R が計算される (ステップ120)。続いて、誤差 e_R としきい値 ϵ' との比較が行われて (ステップS122)、 $|e_R| < \epsilon'$ であれば、処理は、ステップS118に復帰する。なお、ここで、 ϵ' である。 10

【0149】

【数35】

$$e_R = \text{Re}\{y_s(k)\} - \sum_{i=1}^n \alpha_R(i) K\{V_{-z}(k), V_{-x}(p_i)\} \quad (28)$$

$$|e_R| < \epsilon'$$

一方、 $|e_R| < \epsilon'$ であれば、以下の式(29)~(30)にしたがって、係数 $\alpha_R(1), \dots, \alpha_R(n)$ 、および誤差 e_R のそれぞれについて、符号を計算し、符号ベクトルを置き換え (ステップS124)、 $[\alpha_R(1), \dots, \alpha_R(n), 0]$ を新たに係数ベクトルに置き換えるとともに、 $(n+1)$ を新たに n と置き換える (ステップS126)。 20

【0150】

【数36】

$$\hat{S}_R \leftarrow [\text{sign}(\alpha_R(1)), \dots, \text{sign}(\alpha_R(n)), \dots, \text{sign}(\alpha_R')]^T \quad (29)$$

$$\alpha_R \leftarrow [\alpha_R(1), \dots, \alpha_R(n), 0]^T \quad (30)$$

$$n \leftarrow n+1$$

($V_{-z}(k), \text{Re}\{y_s(k)\}$) を n 番目のサポートベクトル ($V_{-z}(P_n), \text{Re}\{y_s(P_n)\}$) として採用して、処理はステップS104に復帰する (ステップS128)。

【0151】

以上の処理により、係数更新部42は、逐次適応的に係数ベクトルの更新を行う。

【0152】

[サポートベクトルマシンSVM54, 58の動作説明]

SVM54は、係数更新部42で更新された係数 $\alpha_R(i)$ を用いて、上述した式(8)に従い、ベースバンド信号 $x(t)$ をベースバンド信号 $\text{Re}\{y(t)\}$ に変換する。サポートベクトルマシンSVM58は、上述したと同様の手続きで、係数更新部66で更新された係数 $\alpha_I(i)$ を用いて、上述した式(9)に従い、ベースバンド信号 $x(t)$ をベースバンド信号 $\text{Im}\{y(t)\}$ に変換する。 40

【0153】

[逐次計算における特徴的手続きの説明]

以上、図5および図6により、サポートベクトルマシンを用いて、非線形なプロセスであって、当該プロセスを表現する変数が複素数で表される場合に、回帰問題を解くことにより、前置型非線形歪補償システムを実現するための構成について説明した。特に、上記係数更新部42の行う処理において、特徴的な手続きについて、以下に説明を付け加えておく。

【0154】

30

50

図7は、1入力1出力の非線形回帰 $f : x \rightarrow y$ に関して、サポートベクトルマシンによる近似学習がある程度進んだ状態において、サポートベクトルによる学習データの近似特性を示す概念図である。

【0155】

図7においては、「損失」は、不感帯2次損失関数に基づく近似特性を表しているものとする。つまり、「損失」は、不感帯の片側幅を近似特性に加減して得られた2つの不感帯境界値のうち、学習データに近い方の不感帯境界値を減算することにより得られる値と定義する。

【0156】

図7で、2重丸はサポートベクトルであるデータを表しており、1重丸は、サポートベクトルによる近似特性に対する不感帯の内側に存在し、サポートベクトルではないデータを表す。なお、サポートベクトルの定義により、図7中で2重丸で表されるデータが真のサポートベクトルであるときは、この真のサポートベクトルの外側には、サポートベクトルは存在しない。

10

【0157】

ここで、たとえば、サポートベクトルによる近似特性の上側であって、かつ、不感帯の外側にあるサポートベクトルのデータ $D1 = (x_1, y_1)$ について見てみると、このデータ $D1$ についての損失 $L1$ は、以下の式で表される。

【0158】

$$L1 = \alpha_1 / C$$

20

なお、ここで、 α_1 はサポートベクトル x_1 に対する係数である。このとき、データ $D1$ の損失は、このデータがサポートベクトルによる近似特性の上側にあることに対応して、「上向き、かつ、外向き」であり、 $\alpha_1 > 0$ である。

【0159】

一方、たとえば、サポートベクトルによる近似特性の下側であって、かつ、不感帯の外側にあるサポートベクトルのデータ $D2 = (x_2, y_2)$ について見てみると、このデータ $D2$ についての損失 $L2$ も、 $L2 = \alpha_2 / C$ で表される。

【0160】

このとき、データ $D2$ の損失は、このデータがサポートベクトルによる近似特性の下側にあることに対応して、「下向き、かつ、外向き」であり、 $\alpha_2 < 0$ である。

30

【0161】

(ステップS122における α の値の設定)

α の値は、本来の不感帯を定義する値 α_0 と同じ値としても計算を実行することは可能である。しかし、本願発明では、 α の値を本来の不感帯を定義する値 α_0 よりも大きくしている。

【0162】

そこで、以上の準備のもとに、まず、図6のステップS122において、 α の値を本来の不感帯を定義する値 α_0 よりも大きくする理由について、まず、説明する。

【0163】

図8は、このように α の値を本来の不感帯を定義する値 α_0 よりも大きくすることの効果

40

を説明するための概念図である。

【0164】

なお、 α で決まる本来の不感帯に対し、 α をステップS122の α で置換えた場合に、 α で決まる領域を本来の不感帯に対する「拡張不感帯」と定義する。

【0165】

拡張不感帯はステップS122のみにおいて用いられ、他のステップにおける不感帯は、 α_0 で決定される。

【0166】

図8においては、学習途中のサポートベクトルによる近似特性が直線 LSV で表され、この近似特性よりもさらに近似が進んだ状態の近似特性が曲線 CSV で表されているものと

50

する。ただし、「学習途中のサポートベクトルによる近似特性」を直線としたのは、説明の便宜のためであって、より一般には、この特性も曲線で表される。

【0167】

まず、近似特性 L S V について、拡張不感帯をしきい値 θ ($= \theta_0$) で定義した場合は、データ D 3 は、最終的にはサポートベクトルにはならないデータであるものの、近似特性 L S V の段階では、サポートベクトルとして選択されることになる。これは、拡張不感帯を定義するしきい値を、しきい値 θ よりも大きなしきい値 θ_1 ($\theta_1 > \theta$) で定義した場合も同様である。

【0168】

これに対して、近似特性 L S V について、拡張不感帯をしきい値 θ_1 ($\theta_1 > \theta$) よりもさらに大きなしきい値 θ_2 ($\theta_2 > \theta_1$) で定義した場合は、データ D 3 は、近似特性 L S V の段階でも、サポートベクトルとして選択されない。

10

【0169】

上述したとおり、本発明においては、計算量を低減するために、逐次計算の途中において、サポートベクトルとして選択されなくなったデータを計算の対象から除外している。

【0170】

したがって、図 6 のステップ S 1 2 2 において使用するしきい値 θ を、予め本来の不感帯に対応するしきい値 θ_1 よりも大きなしきい値 θ_2 としておけば、計算が進むにつれて、除外されていくはずのデータをそもそもサポートベクトルとして選択する確率が小さくなるので、一層の計算量の低減を期待できる。

20

【0171】

すなわち、1 度、サポートベクトルとして抽出した古いデータセットが、学習（新しいサポートベクトルの追加）が進むにつれ、サポートベクトルとして不要となり除去されることが繰り返され、最終的には新しいサポートベクトルの追加が不要となって学習が終了することになる。しかしながら、「サポートベクトルの追加・除去の繰り返し」を低減することにより、計算量を削減するとともに学習の終了までの時間を短縮することが可能となる。

【0172】

なお、新しいサポートベクトルの追加の頻度が低下するにつれてしきい値 θ (θ_0) を θ_1 に近づけていくという処理を行うことも可能である。この場合、ある程度の近似精度までは早期に学習することになり、かつ、最終的な精度 θ_1 も確実に得ることが可能となる。

30

【0173】

また、係数更新部 4 2 や係数更新部 6 6 において、直交検波器 2 2 から受取る学習データの数をカウントするカウンタを備えて、一定数のデータを受取るごとにしきい値 θ を更新しつつ、 θ_1 に近づけていくという処理を行うことも可能である。

【0174】

あるいは、係数更新部 4 2 や係数更新部 6 6 において、時間を計測するカウンタまたは抽出されたサポートベクトルの数をカウントするカウンタを有し、処理期間の長さに対応して、しきい値 θ を更新しつつ、 θ_1 に近づけていくという処理を行うことも可能である。

【0175】

(サポートベクトルを削除する基準)

さらに、図 5 のステップ S 1 1 0 から S 1 1 4 において、サポートベクトルを削除する基準について、さらに説明する。

40

【0176】

図 9 は、図 8 で示される状態から、近似特性の逐次計算を進めた場合に、余分となったサポートベクトルの除去を行う手続きを説明するための概念図である。

【0177】

以下の説明では、図 8 で示される状態で、しきい値 θ と近似特性 L S V により決まる不感帯の外にデータ D 3 があるため、データ D 3 がサポートベクトルとして選択されており、図 9 に示す状態に近似を進めると、このデータ D 3 がサポートベクトルから除去され、一

50

方、データD1が新たにサポートベクトルとして追加されるものとする。

【0178】

図10は、図9において、データD3について、損失の変化を説明するための概念図である。

【0179】

なお、ここでは、直線LSVで表される近似特性で、このデータD3に対応する係数を $\alpha_i(1)$ とし、曲線CSVで表される近似特性でのデータD3に対応する係数を $\alpha_i(2)$ と表すものとする。

【0180】

直線LSVの不感帯の境界の外側にデータD3が存在することに対応して、直線LSVに対するデータD3の損失は、 $\alpha_i(1)/C(\alpha_i(1) > 0)$ となる。

10

【0181】

これに対して、曲線CSVの不感帯の境界の内側にデータD3が存在することに対応して、曲線CSVに対するデータD3の損失は、 $\alpha_i(2)/C(\alpha_i(2) < 0)$ となる。

【0182】

したがって、近似が進んで、サポートベクトルによる近似特性が、直線LSV(より一般的には曲線)から、曲線CSVへと変化することによる損失の変化量を、 $-\alpha_i/C$ とすると、ステップS108で用いたステップサイズは、 $= -\alpha_i(1)/\alpha_i$ と表される。

【0183】

20

このとき、 $0 < 1$ が成り立っていれば、図10で示すように、直線LSV(より一般的には曲線)の不感帯の境界と曲線CSVの不感帯の境界との間にデータD3が存在することになる。これは、言い換えると、「データD3は、サポートベクトルによる近似特性が、直線LSV(より一般的には曲線)から、曲線CSVと変化することにより、サポートベクトルから除去されるデータとなった」ということができる。なお、以上の説明では、データが近似特性の上側にある場合について説明したが、データが近似特性の下側にある場合にも同様である。

【0184】

したがって、図5のステップS108におけるような基準で判断することにより、それまでサポートベクトルとして選択していたデータをサポートベクトルから除外するべきか否かを判断することが可能となる。

30

【0185】

上述の例では、入出力特性が不明な、非線形の入出力特性を有する増幅器16の出力の補正に対して、サポートベクトルマシンを用いたが、本発明は増幅器16に対してのみ適用可能なわけではない。一般的に、入力される信号が複素数で表現でき、かつ入出力特性が不明な信号の変換回路に対しても本発明を適用することが可能である。

【0186】

以上説明したように、前置型非線形歪補償システム1000によると、SVM26およびSVM62を用いて、増幅器16の逆特性を有する信号を推定する。その信号とSVM54およびSVM58より出力される信号とに基づいて、係数更新部42および係数更新部66において、増幅器16の逆特性を有する関数の係数が更新される。

40

【0187】

この構成では、ルックアップテーブルを必要としないため、ノイズフロアを低く押さえることができ、かつ、入力が複素信号である場合に適用可能であるので、簡単な構成で前置型非線形歪補償器を提供することができる。また、サポートベクトルマシンは、*pth-order-predictor*に比べて、計算が簡単で収束時間が早い。さらに、学習する過程で、一度、サポートベクトルに加えられた信号ベクトルにおいて、不要となったサポートベクトルが削除されるため、高速処理が可能な前置型非線形歪補償器を提供することができる。

【0188】

50

また、広帯域信号が入力された場合、増幅器の非線形性が信号帯域に比較して狭帯域の場合にもデータベクトルのサイズ m を適切に設定することにより補償することが可能である。

【0189】

[本発明の実施の形態の前置型非線形歪補償システム2000]

以上、図4に示した前置型非線形歪補償システム1000の構成および動作について説明した。

【0190】

前置型非線形歪補償システム1000は、上述のような特徴を有するものの、図20に示したような入出力特性を有する増幅器において、バックオフを小さく取ると以下のような問題がある。

10

【0191】

すなわち、増幅器への入力が大きき場合、増幅器の飽和特性により、出力はある値に抑えられる。その飽和した出力が推定用SVM26および62の入力となるので、推定用SVM26および62の出力は変換用SVM54および58の出力に比べて小さくなる。

【0192】

したがって、推定用SVM26および62の出力と変換用SVM54および58の出力との差分は、入力信号が小さいときには非常に小さな値となるが、入力が大ききときは非線形歪の影響を受け大きくなる。

20

【0193】

図4に示した前置型非線形歪補償システム1000の方式では、各トレーニングデータについて、両SVM出力の差分が予め設定した不感帯よりも大ききとき、サポートベクトルとして採用する。過剰に大きな振幅のトレーニングデータにおける両SVM出力の差分は、確実に不感帯よりも大きくなり、サポートベクトルとして採用される。このため、このトレーニングデータにより全てのサポートベクトルは更新され、回帰関数の近似精度は悪くなる。このようなトレーニングデータが複数個採用されることにより、回帰関数の精度はますます悪くなり、場合によっては発散してしまう。

【0194】

図11は、本発明の前置型非線形歪補償システム2000の構成を説明するための概略ブロック図である。

30

【0195】

図4に示した前置型非線形歪補償システム1000に比べて、前置型非線形歪補償システム2000には、データ選定部70がさらに設けられている。前置型非線形歪補償システム2000のその他の構成は、前置型非線形歪補償システム1000と同様であるので、同一部分には同一符号を付して、その説明は繰り返さない。

【0196】

なお、図11において、非線形デバイス16は、たとえば、図20に示したような入出力特性を有する増幅器であるものとする。

【0197】

図11を参照して、データ選定部70は、入力されるベースバンド信号 $x(t)$ と、後に説明するしきい値 TH とを比較し、信号 $x(t)$ がしきい値よりも大きき場合、その信号 $x(t)$ を含むデータセットをサポートベクトルとして採用しないようにするものである。

40

【0198】

図12は、データ選定部70の構成を説明するための概略ブロック図である。

図12を参照して、データ選定部70は、ベースバンド信号 $x(t)$ を伝達する経路を選択するための選択部702と、少なくとも指定された個数の N_{TH} 個のベースバンド信号 $x(t)$ を選択部702から受けて格納するためメモリ704と、メモリ704中に格納されたベースバンド信号 $x(t)$ からしきい値 TH を算出するためのしきい値決定部706と、選択部702からのベースバンド信号 $x(t)$ としきい値 TH とを比較して、しきい

50

値 T_H 以下のベースバンド信号 $x(t)$ を選択的に係数更新部 42 および 66 に与えるための比較部 708 とを備える。

【0199】

ここで、しきい値決定部 706 は、以下の式に基づいて、しきい値 T_H を算出する。

【0200】

【数37】

$$T_H = K_1 \times \frac{1}{N_{TH}} \sum_{t=0}^{N_{TH}-1} |x(t)|$$

10

ここで、上式において、 K_1 および N_{TH} は、予め指定される定数である。

【0201】

なお、しきい値 T_H を計算するためのベースバンド信号 $x(t)$ は、たとえば、一定の時間間隔ごとに少なくとも個数 N_{TH} がメモリ 704 に格納されて、しきい値決定部 706 が、この一定時間間隔ごとにしきい値 T_H を更新することとしてもよい。

【0202】

また、しきい値 T_H の決定の仕方は、上述した方法に限られず、たとえば、所定の定数としてもよいし、あるいは、以下の式のように、過去のベースバンド信号 $x(t)$ の絶対値の最大値に所定の定数 K_2 を乗算することで、更新することとしてもよい。

20

【0203】

【数38】

$$T_H = K_2 \times \max_t |x(t)|$$

すなわち、データ選定部 70 においてベースバンド信号 $x(t)$ を選択するためのしきい値 T_H は、外部から任意に設定可能である。または、データ選定部 70 におけるしきい値 T_H は、過去の入力信号から、所定の計算により設定もしくは更新することが可能である。

30

【0204】

次に、図 11 に示した前置型非線形歪補償システム 2000 について説明する。ただし、その動作は、基本的に、図 5 および図 6 で説明した前置型非線形歪補償システム 1000 の動作と同様であるので、以下では、主として、前置型非線形歪補償システム 2000 の動作の前置型非線形歪補償システム 1000 の動作との相違点について説明する。

【0205】

図 13 および図 14 は、前置型非線形歪補償システム 2000 の係数更新の動作を説明するためのフローチャートである。

【0206】

まず、図 13 を参照して、サポートベクトル係数の第 1 次近似を行うため、データセットを N セット取得する。その際、対応するベースバンド信号 $x(t)$ も併せて取得しておく（ステップ S200）。

40

【0207】

次に、データ選定部 70 において、しきい値 T_H を決定する（ステップ S201）。ここでは、たとえば、上述したうち、個数 N_{TH} のベースバンド信号 $x(t)$ の絶対値の平均値に係数 K_1 を乗じて得られるものとする。

【0208】

次に、各データセットについてベースバンド信号 $x(t)$ としきい値 T_H とを比較し、入力信号がしきい値 T_H よりも大きい場合は、そのデータを含むデータセットは回帰関数の精度を悪くするデータセットであるとみなし、サポートベクトルに採用しない（ステップ

50

S 2 0 2)。

【 0 2 0 9 】

残りのデータセットの数を n とすると、データセットは、 $(V_z(P_i), Re\{y_s(P_i)\})$ ($i = 1 \sim n$) である。係数更新部 4 2 は、上記サポートベクトルに対応する係数の第 1 次近似と、その符号ベクトル S_R を、図 5 におけるステップ S 1 0 2 と同様にして計算する (ステップ 2 0 3)。

【 0 2 1 0 】

以下、図 5 のステップ S 1 0 4 ~ S 1 1 6 と同様の処理を、図 1 3 においても、ステップ S 2 0 4 ~ S 2 1 6 として実行する。

【 0 2 1 1 】

次に、図 1 4 を参照して、新しい学習データセット $(V_z(k), Re\{y_s(k)\})$ の取得が行われる (ステップ S 2 1 8)。併せて、対応する入力信号のベースバンド信号 $x(t)$ も取得する。

【 0 2 1 2 】

そのときの入力信号のベースバンド信号 $x(t)$ がしきい値 TH 以上である場合 (ステップ S 2 1 9)、そのデータセットをサポートベクトルに採用しないで、処理は、ステップ S 2 1 8 に復帰する。

【 0 2 1 3 】

一方、ステップ S 2 1 9 において、入力信号のベースバンド信号 $x(t)$ がしきい値 TH よりも小さい場合は、さらに、上述した式 (2 8) により、誤差 e_R が計算される (ステップ 2 2 0)。

以下、図 6 におけるステップ S 1 2 2 ~ 1 2 8 と同様の処理が、図 1 4 においても、ステップ S 2 2 2 ~ S 2 2 8 として実行される。

【 0 2 1 4 】

以上の処理により、係数更新部 4 2 は、逐次適応的に係数ベクトルの更新を行う。

【 0 2 1 5 】

SVM 5 4 は、係数更新部 4 2 で更新された係数 $r(i)$ を用いて、上述した式 (8) に従い、ベースバンド信号 $x(t)$ をベースバンド信号 $Re\{y(t)\}$ に変換する。サポートベクトルマシン SVM 5 8 は、上述したのと同様の手続きで、係数更新部 6 6 で更新された係数 $r(i)$ を用いて、上述した式 (9) に従い、ベースバンド信号 $x(t)$ をベースバンド信号 $Im\{y(t)\}$ に変換する。

【 0 2 1 6 】

[シミュレーション結果]

以下、本発明の効果を示すために、シミュレーション結果を示す。

【 0 2 1 7 】

以下のシミュレーションでも、NLD は増幅器とし、その入出力特性は次式で表されるものとする。

【 0 2 1 8 】

【数 3 9】

$$|S_{OUT}| = \frac{|S_o|}{(1 + |S_o|^6)^{1/6}}$$

$$\angle(S_{OUT}) = \frac{\pi}{3} \frac{|S_o|^2}{1 + |S_o|^2}$$

その他の条件として、入力信号は OFDM 信号とし、そのキャリア数は 1 4 0 5 であり、

10

20

30

40

50

FFTサイズは2048であるものとする。

【0219】

信号電力対雑音電力比(以下、「SNR」という)は、100(dB)である。入力電力は1とし、しきい値THは固定で0.8としている。

【0220】

データセット数は2000である。また、 $m = 1$ とする。さらに、 d は既知であるものとする。

【0221】

図15は、入力信号をバックオフ6(dB)で、増幅器16に通した後の信号スペクトラムである。すなわち、図15においては、前置型非線形歪補償システムを設けることなく、増幅器16で、入力信号を増幅した場合の特性のシミュレーション結果を示している。

10

【0222】

図15において、横軸は周波数キャリア番号、縦軸は電力を表している。非線形歪の影響を受けて、SNRは、約20(dB)にまで悪化している。

【0223】

さらに、図16は、このような状況において、データ選定部70が設けられていない前置型非線形歪補償システム1000による信号スペクトラムを示す。

【0224】

増幅器16のみの場合よりも信号帯域外のスペクトラムが上昇しており、逆にSNRは小さくなってしまふ。これは、過剰に大きな信号を要素にもつデータセットがサポートベクトルとして採用されてしまったからである。そのデータセットにより回帰関数は、もはや増幅器の逆特性を正確に近似できるものではなくなっている。そのデータセット以降のデータセットは、すべてサポートベクトルとして採用されることになり、サポートベクトル数が多くなる。併せて、サポートベクトル係数は極端に大きな数字となり、やがて発散する。

20

【0225】

図17は、図16の状態におけるサポートベクトル係数を表す図である。図17において、横軸はサポートベクトル番号、縦軸はサポートベクトル係数値である。このとき、サポートベクトル数は約1450個と非常に多く、それに伴って計算量も莫大なものになっている。

30

【0226】

図18は、データ選定部70を搭載した本発明の前置型非線形歪補償システム2000による信号スペクトラムを示す図である。増幅器16のみの場合よりも信号帯域外のスペクトラムが下降しており、SNRは約30(dB)に改善された。これは、発散を招く原因となる過剰に大きな信号を要素にもつデータセットをサポートベクトルに採用しなかったことが理由であると考えられる。

【0227】

したがって、バックオフが小さい場合でも、本発明による方式により非線形歪の影響を小さくすることができる。

【0228】

図19は、図18におけるサポートベクトル係数を表す図である。サポートベクトル数は71個となっており、前置型非線形歪補償システム1000と比較して大幅にサポートベクトル数が減少していることが確認できる。併せて、カーネル行列の大きさが小さくなったことにより、計算量も格段に減少している。

40

【0229】

今回開示された実施の形態はすべての点で例示であって制限的なものではないと考えられるべきである。本発明の範囲は上記した説明ではなくて特許請求の範囲によって示され、特許請求の範囲と均等の意味および範囲内でのすべての変更が含まれることが意図される。

【0230】

50

【発明の効果】

以上説明したとおり、本発明によれば、サポートベクトルマシンを用いて、増幅器等の信号の変換回路の逆特性を有する信号を推定するので、ルックアップテーブルを必要としないため、ノイズフロアを低く押さえることができる。

【0231】

しかも、入力が複素信号である場合に適用可能であるので、簡単な構成で前置型非線形歪補償器を提供することも可能となる。

【0232】

さらに、学習する過程で、一度、サポートベクトルに加えられた信号ベクトルにおいて、不要となったサポートベクトルが削除されるため、高速処理が可能である。

10

【0233】

また、本発明により、SVRのみを用いた前置型非線形歪補償器歪においては、補償できない過剰に大きな信号をトレーニング用のデータセットから削除するので、回帰関数の精度を維持することができる。

【0234】

さらに、本発明により、データセットを削除したことにより計算が安定し、同時にサポートベクトル数を少なくすることができるため、計算量を大幅に削減することが可能となる。

【図面の簡単な説明】

【図1】 線形 不感損失関数および二次 不感損失関数を示す図である。

20

【図2】 線形近似関数とサポートベクトルとの関係を示す概念図である。

【図3】 非線形近似関数とサポートベクトルを示す概念図である。

【図4】 本発明の前置型非線形歪補償システム1000の構成を説明するための概略ブロック図である。

【図5】 係数の更新の具体的な計算方法を説明するための第1のフローチャートである。

【図6】 係数の更新の具体的な計算方法を説明するための第2のフローチャートである。

【図7】 サポートベクトルマシンによる近似学習がある程度進んだ状態において、サポートベクトルによる学習データの近似特性を示す概念図である。

30

【図8】 γ の値を本来の不感帯を定義する値 γ_0 よりも大きくすることの効果の説明するための概念図である。

【図9】 近似特性の逐次計算を進めた場合に、余分となったサポートベクトルの除去を行う手続きを説明するための概念図である。

【図10】 図9において、データD3について、損失の変化を説明するための概念図である。

【図11】 本発明の前置型非線形歪補償システム2000の構成を説明するための概略ブロック図である。

【図12】 データ選定部70の構成を説明するための概略ブロック図である。

【図13】 前置型非線形歪補償システム2000の係数更新の動作を説明するための第1のフローチャートである。

40

【図14】 前置型非線形歪補償システム2000の係数更新の動作を説明するための第2のフローチャートである。

【図15】 入力信号をバックオフ6(dB)で、増幅器16に通した後の信号スペクトラムである。

【図16】 データ選定部70が設けられていない前置型非線形歪補償システム1000による信号スペクトラムを示す。

【図17】 図16の状態におけるサポートベクトル係数を表す図である。

【図18】 データ選定部70を搭載した本発明の前置型非線形歪補償システム2000による信号スペクトラムを示す図である。

50

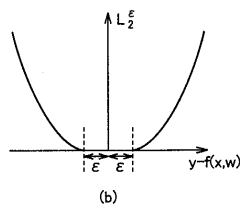
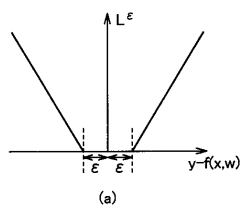
【図19】 図18におけるサポートベクトル係数を表す図である。

【図20】 一般的な増幅器の入出力特性を示す図である。

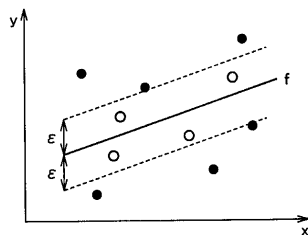
【符号の説明】

2, 18 ダウンコンバータ、4, 20 A/D変換器、6, 22 直交検波器、54, 58, 26, 62 SVM、10 直交変調器、12 D/A変換器、14 アップコンバータ、16 増幅器、30, 64 減算器、42, 66 係数更新部、70 データ選定部、1000, 2000 前置型非線形歪補償システム。

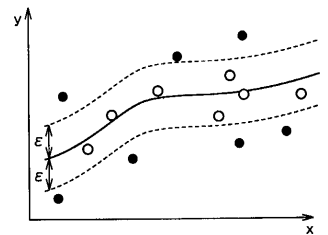
【図1】



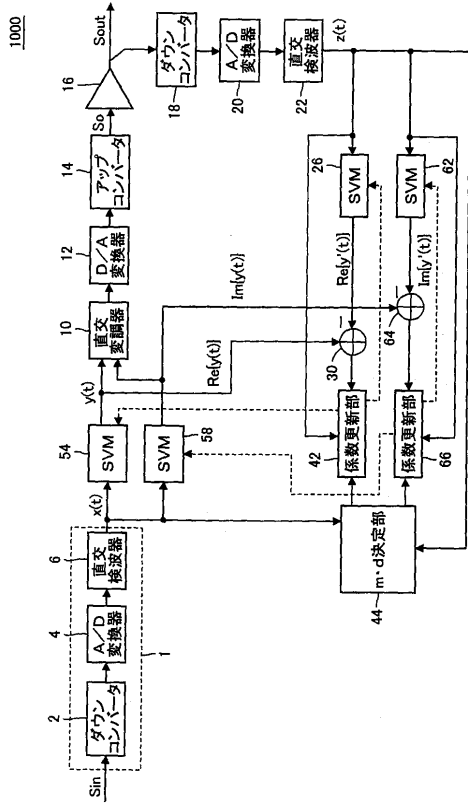
【図2】



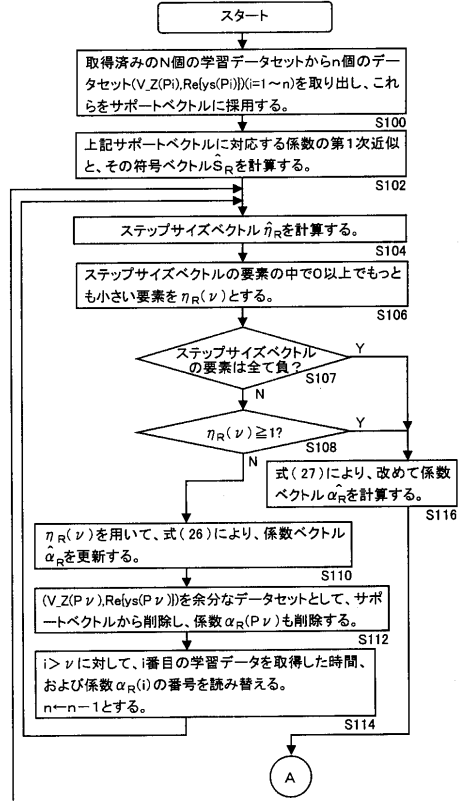
【図3】



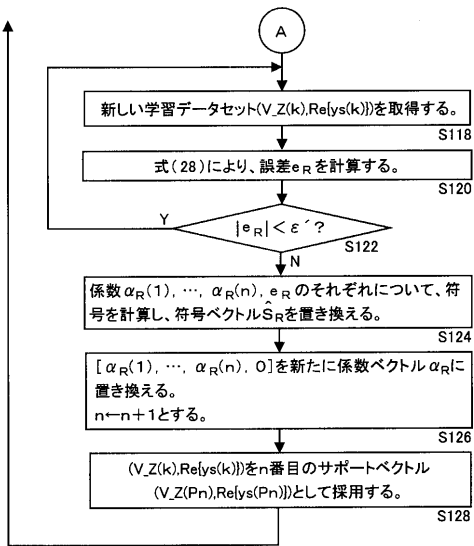
【図4】



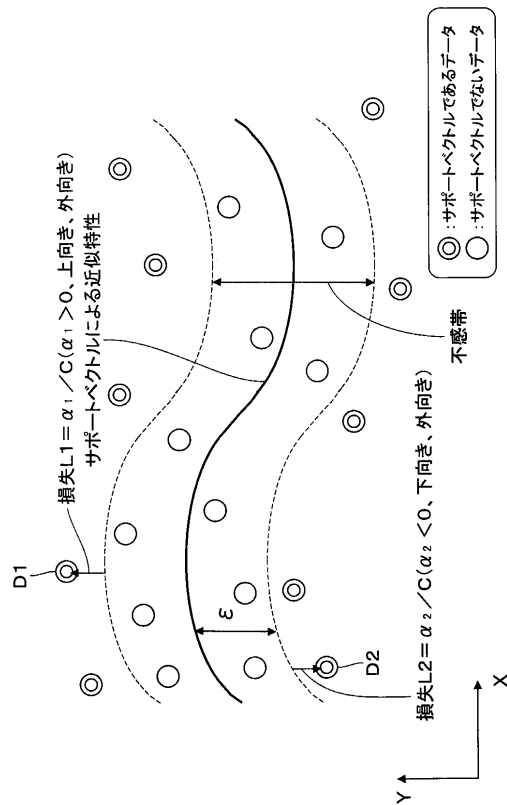
【図5】



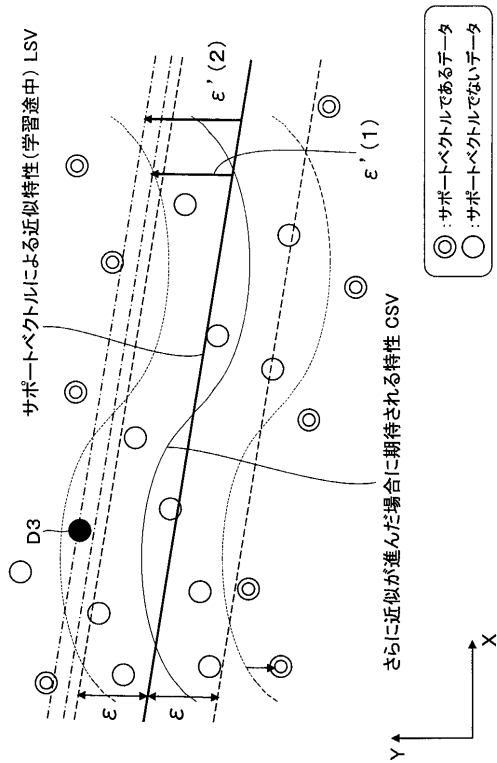
【図6】



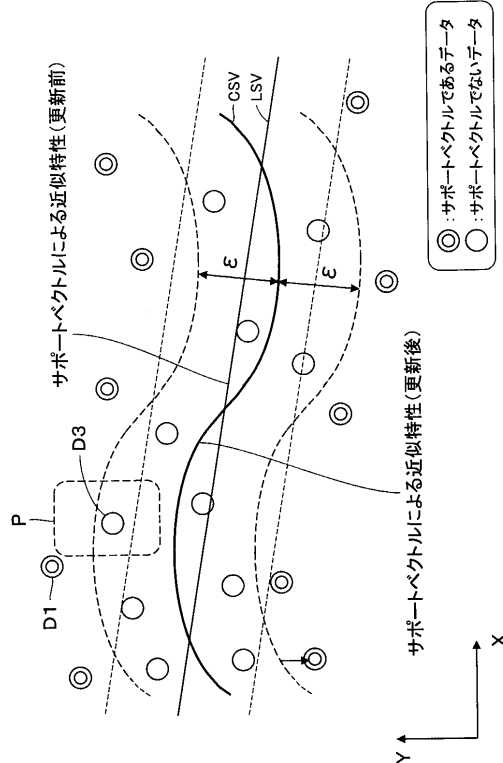
【図7】



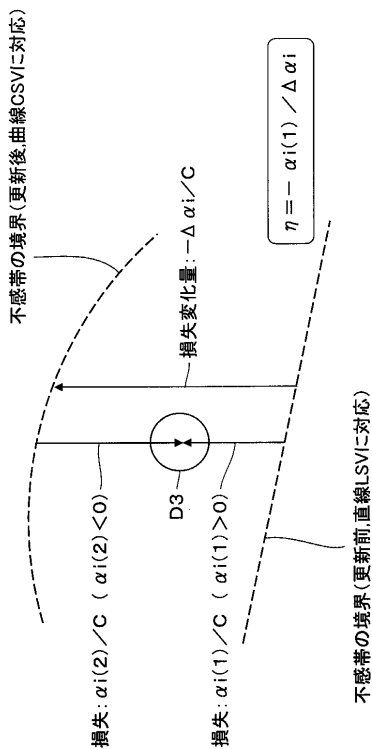
【 図 8 】



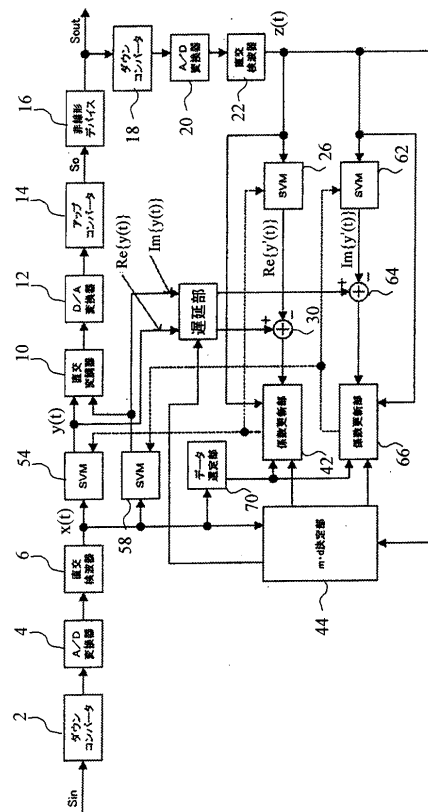
【 図 9 】



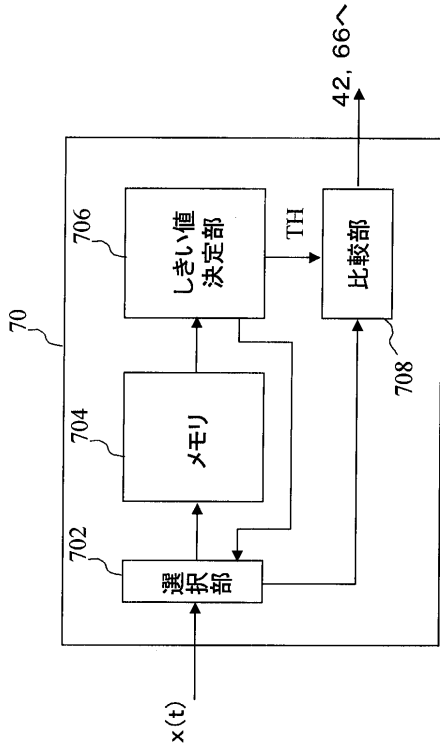
【 図 10 】



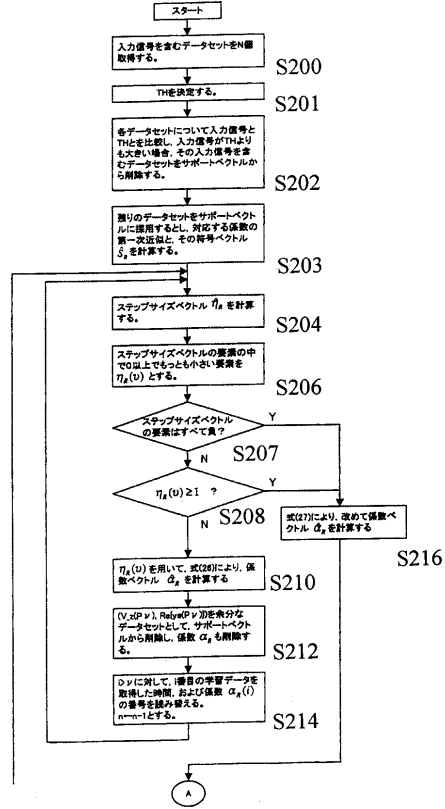
【 図 11 】



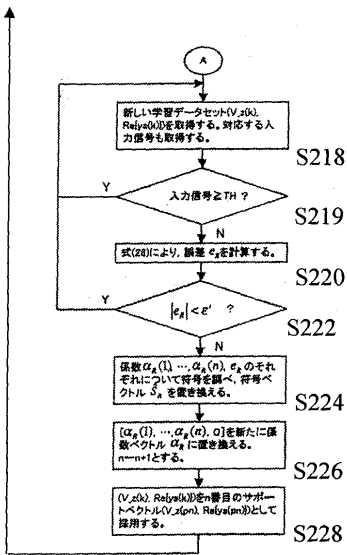
【図12】



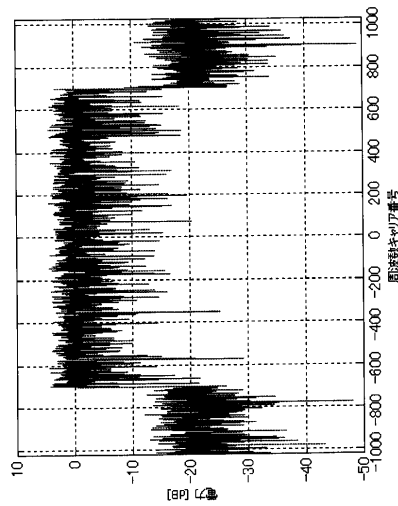
【図13】



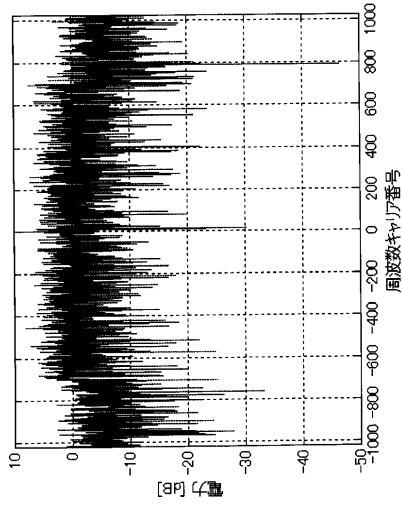
【図14】



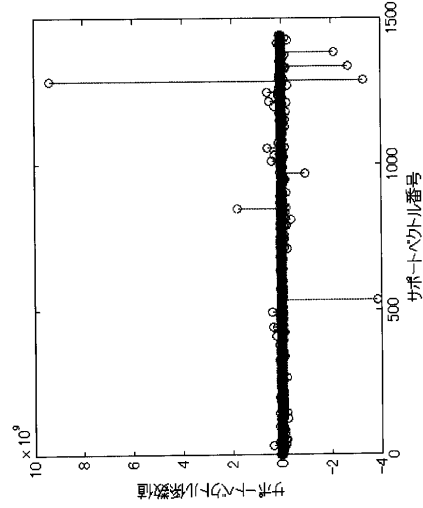
【図15】



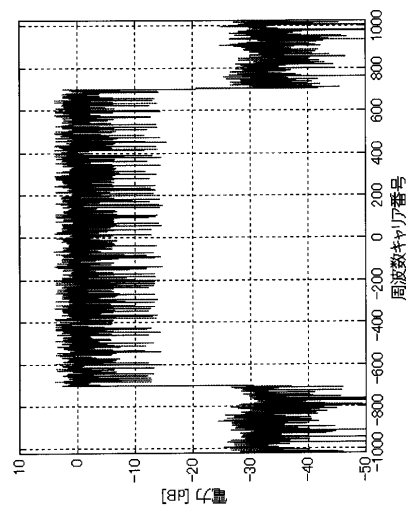
【図16】



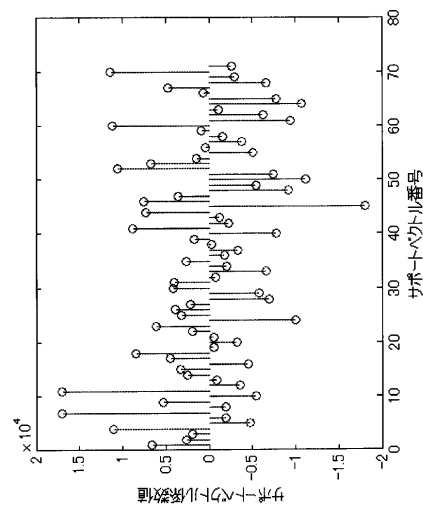
【図17】



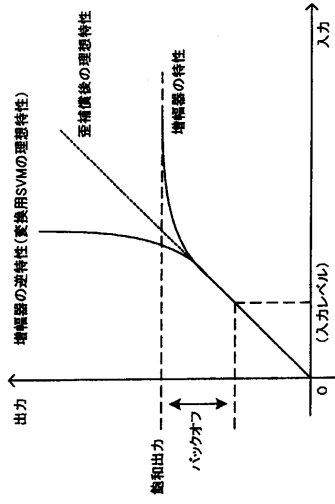
【図18】



【図19】



【 20 】



フロントページの続き

(72)発明者 平川 満

大阪市此花区島屋一丁目1番3号 住友電気工業株式会社 大阪製作所内

審査官 儀同 孝信

(56)参考文献 特開2002-261848(JP,A)

特開2001-251148(JP,A)

特開2001-284976(JP,A)

特開2001-268150(JP,A)

(58)調査した分野(Int.Cl., DB名)

H03F 1/00- 3/45、 3/50- 3/52、

3/62- 3/64、 3/68- 3/72、

H04B 1/76- 3/44、 3/50- 3/60、

7/005- 7/015、

H04L 27/00-27/38