ニューラル・ネットによる潜水艇の運動制御

——水平航行の制御への適用——

正員 藤 井 輝 夫* 正員 浦 環**

Control with Neural Network For Autonomous Underwater Vehicle —Application To Longitudinal Motion Control—

by Teruo Fujii, *Member* Tamaki Ura, *Member*

Abstract

In order to enhance the advantage of the autonomous underwater vehicle (AUV), it is necessary to construct a control system which is capable of letting the vehicle swim in the unknown underwater environment. This paper introduces the neural network as a robust and adaptable control system for this purpose.

Two procedures with linear feedback control law and Fuzzy control algorithm are carried out to make teaching samples for a cruising type underwater vehicle "PTEROA" through numerical simulations of its motion. Two neural networks are structured by several ten thousand times of updates of their weights with reference to these samples.

Both of the networks show good performance in maneuvering a small model of "PTEROA" in a circulating water tunnel in spite of considerable amount of disturbances such as ununiform flow patterns. It should be noted that the original Fuzzy controller failed to control it adequately.

It is concluded that the neural network has vast potential for autonomous control systems of underwater vehicles.

1. はじめに

海中または海底において調査あるいは探索を広域にわたって行うには、通信の乏しさ等の理由から、その手段として自律型潜水艇(AUV: Autonomous Underwater Vehicle)を考えざるを得ない¹⁾。AUVでは、機器を搭載可能なスペースや電力源の限界等の理由で、大規模なシステムによるサポートは困難であるため、その運動制御系は、艇内に搭載した比較的小さいシステムによって実現でき、しかも高い自律性を有する必要がある。

近年、盛んに研究が行われているニューラル・ネット、あるいはその数学モデルは、次の2つの重要な特徴を持つ。

(1) 処理を並列分散的に行うため多くのデータを同時に扱うことが可能で、拡張性に富み、処理結果の信頼性が高い。

(2) 学習機能を有するため、必要な処理を行うネットワークを比較的手軽に構成でき、多くの事例に当りながら

処理の妥当性を向上できる。

これらは、先に述べた特殊な環境条件により、AUVの制御系として好ましい性質である。本論では後者の特徴に着目して、ニューラル・ネットを用いた制御系を構成する。AUVの例として、筆者らが開発中の無索無人潜水艇PTEROA^{2,3)}を制御対象とし、その縦運動の制御に適用する。構成された制御系の有効性を計算機シミュレーションと回流水槽中の実験によって検討する。前者の特徴については、ニューラル・ネットのハードウェア的な実現と大きくかわる部分であるためここでは扱わない。

2. ニューラル・ネットの数学モデル

ニューラル・ネットについて多くの数学モデルが提案されているが⁴⁾、本論では、並列分散処理と学習機能に着目して、PDPモデル⁵⁾(Parallel Distributed Processing Model)を用いる。PDPモデルは、一般に入力層、出力層と何層かの中間層からなる層構造を持ち、各層に数個~数十個のニューロンを含む(Fig. 1 参照)。本論では簡単のため、層内でのニューロン間の結合は考えない。第 n 層目、第 i 番目のニューロンは、次の式に従って出力 x_i^n を与える。

* 東京大学大学院工学系

** 東京大学生産技術研究所

Fig. 1 PDP Model

Fig. 2 Logistic Function

$$u_i^n = \sum_j w_{ij}^{n-1} x_j^{n-1} + h_i^n$$

$$x_i^n = f(u_i^n)$$

$$f(u) = 1 / (1 + \exp(-u)) \quad (1)$$

ここに、 u_i^n ：ニューロンの膜電位、 w_{ij}^{n-1} ：第 $n-1$ 層 j 番目のニューロンから第 n 層 i 番目のニューロンへの結合荷重、 h_i ：しきい値である。 f は入出力関数 (Fig. 2 参照) で、ロジスティック関数と呼ばれる。

この回路網全体の振舞いを決定するのは、各ニューロン素子間の結合荷重 w_{ij}^n である。 w_{ij}^n を変化させることにより、回路網に一定の目的のための処理を行なう機能を持たせる作業を学習という。学習はバックプロパゲーション⁵⁾によって行なう。ある入力に対する出力層からの最終的なネットワーク出力を o_i とし、その入力に対応する望ましい出力を t_i とする。次式で与えられる出力の二乗誤差の和 E を考え、

$$E = \frac{1}{2} \sum_i (t_i - o_i)^2 \quad (2)$$

E を最小 (厳密には極小) とするように w_{ij}^n の値を変化させる。一般化デルタルールに従い、次式によって結合荷重に変化を与える。すなわち、 w_{ij}^n の p ステップ目の変化量を $\Delta w_{ij}^n(p)$ として

$$\Delta w_{ij}^{n-1}(p+1) = \eta \delta_i^n x_j^{n-1} + \alpha \Delta w_{ij}^{n-1}(p) \quad (3)$$

ここで、 δ_i^n は学習信号で次式により与えられる。

$$\delta_i^n = (t_i - o_i) f'(u_i^n) : \text{出力層} \quad (4)$$

$$\delta_i^n = f'(u_i^n) \sum_k \delta_k^{n+1} w_{ki}^n : \text{中間層} \quad (5)$$

以上の式を用いて繰り返し結合荷重を変化させることにより、必要な処理能力を備えたネットワークを実現する。(3) 式の η , α は荷重変化の速度及び慣性を調整するもので、収束に問題がない限りは小さい適当な値に設定しておく。

3. 制御対象

無索無人潜水艇 PTEROA^{2,3)} は、グライディング機能を重視した翼状の胴体形状で、制御面として左右 1 対のエレベータと上下に垂直舵を持つ。本論では実験に供する全長 60 cm の有索自航模型である PTEROA60^{6,7)} を制御対象とする。主要寸法を Table. 1 に外観を Fig. 3 に示す。PTEROA60 は自航のため各種センサ類、アクチュエータ、スラスタを搭載し、電力供給とデータ通信のためのケーブルによって外部のコンピュータとリンクしている (Table. 2 参

Table. 1 Dimensions of PTEROA60

全長	: L	0.6 m
全幅	: B _{0a}	0.39 m (0.65 L)
胴体幅	: B	0.3 m (0.5 L)
胴体厚さ	: T	0.18 m (0.3 L)
排水量	: $\nabla \rho g$	17.45 kg
重量	: W	1.005 $\nabla \rho g$
浮心の位置	: x_a	0.243 m (前端より)
重心の位置	: x_g	0.235 m (前端より)

Fig. 3 PTEROA60 Submersible

Table. 2 Instruments on Vehicle

アクチュエータ	垂直舵用ステップモータ 右舷水平舵用ステップモータ 左舷水平舵用ステップモータ
センサ	ピッチ角用傾斜計 ロール角用傾斜計 深度計測用圧力計 方位計
スラスタ	交流モータ (100V, 15W)

Fig. 4 Learning Systems for Neural Network

Table. 3 Input and Output of Control System

CASE	教師		ニューラル・ネット			
	入 力	出 力	入 力	出 力	中間層の素子数	
A	$\theta - \theta_0, \dot{\theta}, d - d_0, \delta_e - \delta_{e0}$	$\dot{\delta}_e$	$\theta, \dot{\theta}, d - d_0$	δ_e	5	
B	$\Delta d, \Delta \theta$	$\Delta \delta_e$	$\Delta d, \Delta \theta$	δ_e	5	

照)。艇の姿勢角及び深度の制御はエレベータを左右同時に同角度回転させることにより行なう。計算機によるシミュレーションには文献2)の運動方程式及び流体力に関する推定式を用いる。

4. ニューラル・ネットを用いた制御系の構成

PDPモデルを用いて、水平航行させることを目標にして、潜水艇の縦運動に関する制御系を構成するには、適当な操縦パターンを学習させねばならない。学習の手順は、

- 1) 特徴的な操縦例を作る。
- 2) 入出力の組を一定数選択して教示データを作る。
- 3) バックプロパゲーションによる繰り返し計算により結合荷重の変化を行なう。

本論では、操縦例を作るために次の2つの制御系(教師)

A. 深度を考慮した線形フィードバック制御

B. 深度、姿勢角変化を考慮したファジー制御

を採用し (Fig. 4 参照), それぞれ CASE A, CASE B と呼ぶ。それらの入力と出力を Table. 3 に示す。

4.1 線形フィードバック制御を用いた教示データの作成 (CASE A)

PTEROA60 に適用して、良好な結果が得られている⁶⁾線形フィードバック則 (Appendix 参照) により、制御のシミュレーションを行った結果をもとに教示データを作る。目標値は深度 d_0 、水平定常航行状態における姿勢角 θ_0 である。制御系への入力は姿勢角 θ 、姿勢角速度 $\dot{\theta}$ 及びエレベータ角 δ_e のそれぞれ水平定常航行状態からの変化分と、深度 d と目標深度 d_0 との差で、出力はエレベータ角の操作角速度 $\dot{\delta}_e$ である (Fig. 5)。制御則の導出に必要な艇体の運動に関するパラメータは縦運動に関するリミットサイクル等の実験により求めている⁶⁾。制御シミュレーション結果を Fig. 6 に示す。初期値は $\theta=0$ rad, $d=1$ m, $\delta_e=-0.1$ rad で、制御目標は $\theta_0=0.049$ rad, $d=3$ m であり、計算ステップは 0.1 秒とした。

4.2 ファジー制御を用いた教示データの作成 (CASE B)

艇体の運動に関する少ない情報をもとに制御を行なう例としてファジー制御を考える。深度と姿勢角の1秒毎の変

Fig. 5 Untethered Submersible in Free Swimming

Fig. 6 Teaching Sample Derived from Numerical Simulation by Linear Feedback Control

if	$\Delta d(\text{cm})$ is	$\Delta \theta(\text{deg})$ is	then	$\Delta \delta_e(\text{deg})$ is
Rule 1	UP 	NEUTRAL 	$\rightarrow$	DOWN
Rule 2	DOWN 	NEUTRAL 	$\rightarrow$	UP
Rule 3		UP 	$\rightarrow$	DOWN
Rule 4		DOWN 	$\rightarrow$	UP

Fig. 7 Fuzzy Algorithm

Fig. 8 Teaching Sample Derived from Numerical Simulation by Fuzzy Control

化量を考慮して、それぞれの変化を抑えるように1秒ごとに制御動作を行なうファジールゴリズムを用いて教示データを作成する。制御系への入力は深度の変化量 Δd と姿勢角の変化量 $\Delta \theta$ で、出力はエレベータ角の操作量 $\Delta \delta_e$ である。ファジールールは、単純なものを考え、メンバーシップ関数は全て三角形のものとした (Fig. 7 参照)。シミュレーションにおいて水平航行が最小限可能なようにメンバーシップ関数のチューニングを行っている。ルール1, 2に関しては $\Delta d, \Delta \theta$ に関する条件それぞれについての適合度の積を、ルール全体に関する適合度とし、後件部に関しては重心法を用いて制御量を算出する。制御シミュレーション結果を Fig. 8 に示す。初期値は $\theta = 0.2 \text{ rad}$, $d = 1 \text{ m}$, $\delta_e = 0 \text{ rad}$ である。

4.3 ネットワーク構造の決定

以上のような教示データをもとに、それぞれの教師に対応したニューラル・ネットを構成する (Table. 3 参照)。CASE A では入力に、姿勢角、姿勢角速度、深度と深度目標値との差を与え、水平定常航行状態における姿勢角は与えない。出力はエレベータ角とする。従ってニューラル・ネットには水平定常航行時の姿勢角と艇の運動に関する情報が含まれる。教示データの組は200組、学習回数は7万回とした。CASE B では入力は教師と同じにし、出力をエレベータ角にした。この場合にも艇の運動に関する情報がニューラル・ネットに包含される。教示データは200組、学習回数は10万回とした。

5. 動作検証のためのシミュレーション計算

5.1 CASE A

初期値、目標値を教示データ作成に用いた例と同じにしたときのシミュレーション結果を Fig. 9 に示す。深度が0.15 m 程度オーバーしているが、ほぼ、教師となった線形フィードバック則と同等の動作を実現している。次に、深度を3 m から1 m に変化させる操縦をニューラル・ネットに行わせた結果を Fig. 10 に示す。初期値は $\theta = 0.049 \text{ rad}$, $\delta_e = -0.16 \text{ rad}$, $d = 3 \text{ m}$ である。これも深度に0.15 m 程度の誤差が現れているが、ほぼ妥当な制御結果を示している。

5.2 CASE B

初期値を教示データ作成に用いた例と同じにしたときのシミュレーション結果を Fig. 11 に示す。姿勢角に関しては、教師であるファジー制御の結果 (Fig. 8) よりも振動が抑えられ、収束が速くなっている。深度は40秒間で0.2 m 程度増加している。次に、初期値を $\theta = 0 \text{ rad}$ として、制御を行った結果を Fig. 12 に示す。この場合も深度変化が完全に0になっていないが、 θ はほとんど振動せずに収束し

Fig. 9 Numerical Simulation of Control by Neural Network (CASE A-1)

Fig. 10 Numerical Simulation of Control by Neural Network (CASE A-2)

Fig. 12 Numerical Simulation of Control by Neural Network (CASE B-2)

Fig. 11 Numerical Simulation of Control by Neural Network (CASE B-1)

Fig. 13 Experimental Results of Motion Controlled by Neural Network (CASE-A)

ている。

6. 動作検証のための自航試験

学習を行なって得られたニューラル・ネットを用いて、東京大学生産技術研究所千葉実験所内の風路付造波回流水槽で PTEROA 60 を自航させ、これを制御した。回流水槽の航行可能範囲の寸法は全深 1 m、全幅 1.8 m、全長約 4 m、流速は 0~1 m/s (可変) である。水槽の寸法上大きな深度変化を伴う運動をさせることが不可能なため、一定深度航行を制御目標とした。また、流速は 0.7 m/s で一定とした。

実験の場合、計算機上のシミュレーションと異なり、制御性能を低下させるいくつかの理由が考えられる。センシングや制御動作の際の電氣的、機械的なノイズ及び誤差、水流の乱れ等の影響を無視できない。ニューロ・コントロ

ールがその特性を有効に発揮するのは、このように、運動を不安定化する要素の多い場合であると考えられる。

6.1 CASE A

深度の目標値を $d=0.4$ m としたときの制御結果を Fig. 13 に示す。深度誤差は最大 10 cm 程度であり良好な制御結果を得た。実際の問題に対して、良好な結果を示すコントローラの機能を学習することにより、ニューラル・ネットによって同等の制御が可能となることが確認された。

6.2 CASE B

ニューラル・ネットによる制御結果を Fig. 14 に示す。比較のために、ファジー制御による制御結果を Fig. 15 に示す。ファジー制御の方はコントローラ自体が、ある意味で人間の主観により調整されていて、対象の運動に関する情

Fig. 14 Experimental Results of Motion Controlled by Neural Network (CASE-B)

Fig. 15 Experimental Results of Motion Controlled by Fuzzy Algorithm

報は直接的には含まれていない。このためシミュレーションにおいても観察されたように、急激な動作による姿勢角の振動をおこし、10秒付近で水槽の底に衝突した。これに対してニューラル・ネットによる制御の方は、幾分鈍い対応により水平航行状態を保っている。すなわち、制御対象の運動特性に関する情報を前情報として包含するニューラル・ネットを構成することにより、運動を不安定にする要素の多い環境においても柔軟な制御を行うことが可能であることが示された。

7. ニューラル・ネットを用いた制御系構成法に関する検討

ニューラル・ネットを用いた制御系をバックプロパゲー

ションによる学習によって構成するには、

- (1) 制御系への入力と出力
- (2) ニューロンの数と層の構成
- (3) 教示データの内容
- (4) 教示データの作成方法
- (5) 学習回数及び諸係数
- (6) しきい値の取扱い

など多くの検討すべき項目が考えられる。本論では研究の第一段階として、ニューラル・ネットによる制御の実現性及び有効性を示すことを目的としたので、全項目にわたる細かな考察は行わず、実際に制御系を構成するにあたり留意した事項について検討する。

7.1 ネットワークの持つ入出力関係

シミュレーション及び実験において良好な結果が得られたネットワークの持つ入出力間の関係を CASE A について Fig. 16 に、CASE B について Fig. 17 に示す。CASE A は $\dot{\theta}=0$ rad/s の時の深度の目標値との差 $d-d_0$ とエレベータ角 δ_e の関係を θ をパラメータとして表している。例えば、 $\theta=-0.5$ rad の曲線は、 δ_e が小さい方向にシフトしており、全体としても滑らかで妥当な入出力関係が得られている。CASE B は $\Delta\theta$ をパラメータにして Δd とエレベータ角 δ_e との関係を表している。 $\Delta d=0$ m 付近では傾き

Fig. 16 I/O Relations of Neural Network (CASE A)

Fig. 17 I/O Relations of Neural Network (CASE B)

Fig. 18 Numerical Simulation of Control by Neural Network with the Same I/O System as Fuzzy Controller's

が大きい Δd の絶対値が0.2を越えるあたりから傾きは小さくなっており、大きな変動に対しても振動を起こさずにやや緩慢な動作をすることが理解できる。

7.2 制御系への入力と出力

制御系の構成にあたって、入力と出力を何にするかという選択は、その制御系の性質を考える上で非常に重要である。本論では、制御の前情報としてニューラル・ネットに制御対象の運動特性に関する情報を組入れるため、前述のような入出力に決定したが、例えばあるコントローラを忠実にコピーしたネットワークをつくる目的ならば、入出力もそのコントローラと同じ組み合わせにしなければならない。また、潜水艇に関して言えば、前方障害物ソナー等の出力など、艇の状態量でない情報で運動制御に必要なものを入力として取り込むことも考えられる。一例として、CASE Bの出力をエレベータ角の操作量 $\Delta\delta_e$ として、ファジー制御系を忠実に学習させたニューラル・ネットでの制御を行ったシミュレーション結果をFig. 18に示す。この場合、Fig. 8のファジー制御による結果と同様に姿勢角 θ の振動が見られ、Fig. 12で示したニューラル・ネットによる制御結果とは異なっている。

7.3 教示データの作成方法

教示データ作成の際に留意すべき点は、与える入出力関係の値の範囲とその比率である。次の3つの場合についてそれぞれ教示データを作成し、これを学習させることにより得られたニューラル・ネットの特性を比較する。

- 制御対象の釣合状態に対して、片側のみの値（例えば $\theta \geq 0$ のデータのみ）が現れるような操縦例を与える。
- a. に加えて a. と対称な操縦例も与える。
- 釣合状態への収束後の定常状態のデータ比率を多

くする。

a. の例としてはFig. 6の線形フィードバックによる操縦例が挙げられる。この動作例は、オーバーシュートがほとんど見られず、 $\theta \leq 0.05$ rad, $(d - d_0) \geq 0$ の範囲の教示データとなっている。得られたニューラル・ネットはFig. 16に示したように、 $\theta \geq 0.05$ rad, $(d - d_0) \leq 0$ の部分も適当に補完した入出力関係を作っている。

b. の例として a. の教示データに a. と対称的な操縦例を同じ数だけ加えたものを作成した。初期値を $\theta = 0.099$ rad, $d = 3$ mとし目標値を $\theta = 0.049$ rad, $d = 1$ mとした操縦の入出力の組を a. で与えた操縦例に加えた。これを学習することによる、ニューラル・ネットの変化の様子をFig. 19に示す。b. の方が a. より幾分傾きの大きい曲線となっているが、さほど大きな相違はない。このように、制御対象が釣合状態のまわりでそれほど非対称な性質を持たない限り、ニューラル・ネットが適当な補完を行うので、教示データの値の範囲の偏りはそれほど大きく影響しない。

c. の例としてFig. 8のファジー制御によるシミュレーション結果を0～40秒までとり、これによって教示データを作成した。この場合データ全体に対して半分以上が0入力($\Delta\theta = 0, \Delta d = 0$)となっており、学習を行うことによりニューロン間の結合荷重の一樣化が観察された(Table. 4参照)。すなわち、ネットワークは望ましい状態に収束せず、あるニューロンから出力された信号がすべて同じ強さで、次のニューロンに到達してしまうような状態となっている。得られたニューラル・ネットの入出力関係をFig. 20に示す。 $\Delta\theta$ の値に応じて入出力関係が深度に対して大きく変化している。このような状態はしきい値 h_i を学習することにより回避できる。しきい値は、常に1または-1の出力を行うニューロンからの結合荷重と考えることにより、結合荷重と全く同様の方法で学習させることができ、これにより0入力の際にも結合荷重の変化が行われるため一樣化が起こりにくくなる。同じ教示データに関して、しきい

Fig. 19 Variation of I/O Relations with Difference of Teaching Data

Table. 4 An Example of Uniform Weight Data (CASE B)

i	w_{i1}^1	w_{i2}^1	w_{i1}^2
1	0.241446	-0.259156	-0.250441
2	0.241073	-0.258737	-0.249817
3	0.240691	-0.258307	-0.249178
4	0.240299	-0.257868	-0.248526
5	0.239896	-0.257416	-0.247855

Fig. 20 I/O Relations of Neural Network with Uniform Weights

Table. 5 An Example of Weights and Thresholds Data (CASE B)

i	w_{i1}^1	w_{i2}^1	w_{i1}^2	h_{i1}	h_{i2}
1	0.298153	-0.268944	-0.738277	-0.282519	0.345811
2	0.203941	-0.146299	-0.558526	0.257841	—
3	0.129234	-0.078939	-0.375636	0.720020	—
4	-0.943475	-0.011795	0.559558	1.017284	—
5	-0.535307	-0.524159	-0.892155	0.271636	—

値も含めて学習を行った結果得られた結合荷重及びしきい値の組を Table. 5 に示す。

8. おわりに

本論文では、ニューラル・ネットの学習機能を応用して、自律型潜水艇の縦運動制御への適用例を示した。

2種類の教師を採用し、その動作を学習することにより、ニューラル・ネットを構成比較した。シミュレーション計算と実験により、得られたニューラル・ネットの動作確認を行い、

- (1) 教師とほぼ同等の動作が実現される。
- (2) 動作域の変化に対しても(教示していない範囲に

関しても)、柔軟に対応する。

- (3) 前情報として制御対象の運動特性を包含することにより制御性能を向上することが可能である。
- ことを明らかにした。特に実験において、ファジー制御は急激な動作により、満足な結果が得られなかったのに対し、これを学習したニューラル・ネットは良好な制御結果を得た。

さらに学習の過程において留意すべきいくつかの事項について検討を行い、以下の知見を得た。

- (1) ニューラル・ネットはある限られた範囲のデータのみを学習することで、それ以外の入力に対しても、適当な補完を行った出力を与える。
- (2) 入出力項目をうまく選択することにより、対象に関する情報を内部に取り込むことが可能である。
- (3) 学習に際しては、0 入力が多くならないように留意し、特徴的な操縦例を与えればよい。

今後の課題として、基礎的な事項では

- (1) ネットワーク構成(素子数、層数等)
- (2) 学習の収束性
- (3) 学習により得られるニューラル・ネットの特性の評価法
- (4) 層内結合を有するモデルの導入

等に関して、より詳細な検討が望まれる。また、潜水艇への応用面で、

- (1) 実機の航行データに基づく学習
- (2) 入出力情報の拡張
- (3) ハードウェア化

などについて今後研究を進める予定である。

本研究を行うに当たり、東京大学生産技術研究所橋本秀紀講師、同工学部石谷久教授より貴重なご助言を賜った。また、実験及び計算の過程で(株)アーバンブレイン横山正氏、山下智氏には、多大なご協力を頂いた。ここに篤く感謝の意を表します。

参考文献

- 1) 藤井輝夫、浦環：“潜水艇の自律性とニューロコントロールの研究”，生産研究，Vol. 41-9，(1989)
- 2) 浦環，大坪新一郎：“航行型無索無人潜水艇に関する研究(その1)”，日本造船学会論文集，Vol. 162(1987)，pp. 117-124
- 3) 大坪新一郎：“グライダー型潜水艇の可能性に関する研究”，東京大学大学院修士論文(1987)
- 4) 甘利俊一：“神経回路網の数理”，産業図書(1978)
- 5) D. E. Rumelhart, J. L. McClelland and The PDP Research Group: “Parallel Distributed Processing”, The MIT Press (1986)
- 6) 浦環，藤井輝夫：“グライダー型潜水艇の設計に関

する研究(その6)”, 生産研究, Vol. 41-9, (1989)

- 7) 山下智: “有索小型模型を用いた無索無人潜水艇の運動の研究”, 東京農工大学大学院修士論文(1989)

Appendix

深度変化を考慮した縦運動の線形フィードバック制御

ピッチ運動に関しては, 舵角から姿勢角にいたるダイナミクスが水平定常航行状態(以後トリム状態という)まわりで二次系で近似できるとし, k_1 を実験により求める比例係数として次の関係式

$$\dot{\theta} = \omega_y, \quad \dot{\omega}_y = k_1 \delta_e \quad (\text{a.1})$$

で表わす。ここに, θ : トリム状態からの姿勢(ピッチ)角の変化量(本文中では $\theta - \theta_0$), ω_y : ピッチ角速度, δ_e : トリム状態からのエレベータ角の変化量(本文中では $\delta_e - \delta_{e0}$)である。さらに, 制御入力 u をアクチュエータであるパルスモータの動作速度と考えると, 舵とモータ間のギヤ比を k_2 とすると

$$\dot{\delta}_e = k_2 u \quad (\text{a.2})$$

である。これらの関係をマトリクスで表わすと $\mathbf{x}^T = [\theta, \omega_y, \delta_e]$, $\mathbf{u} = [u]$ として

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (\text{a.3})$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & k_1 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ k_2 \end{bmatrix}$$

となる。姿勢角の変化のみで深度を一定に保つため, 制御目標の姿勢角 θ_d は, 次式で決定する。

$$\theta_d = \tan^{-1}\{(d - d_0)/H\} \quad (\text{a.4})$$

ここに, d : 深度, d_0 : 目標深度, H : 深度回復の目安とする前進距離である。 H はピッチ運動の時定数を考えて, 振動を起こさない程度の値に設定する。

以上より, 次の評価関数 J を最小にするような, 制御入力を与えるものとする。

$$J = \int \{q_1(\theta - \theta_d)^2 + q_2\omega_y^2 + q_3\delta_e^2 + ru^2\} dt \quad (\text{a.5})$$

ここに, q_1, q_2, q_3, r は重みパラメータである。制御入力算出のための線形フィードバック則を

$$u = g_1(\theta - \theta_d) + g_2\omega_y + g_3\delta_e \quad (\text{a.6})$$

とすると, あらためて $\mathbf{x}^T = [\theta - \theta_d, \omega_y, \delta_e]$ において

$$\mathbf{u} = \mathbf{G}\mathbf{x}, \quad \mathbf{G} = \text{diag}(g_1, g_2, g_3) \quad (\text{a.7})$$

とあらわせる。また, この $\mathbf{G}$ は, 次に示すリカッチ方程式(a.8)の定常解 $\mathbf{K}$ を(a.9)式に代入して算出される。すなわち

$$-\dot{\mathbf{K}} = \mathbf{K}\mathbf{A} + \mathbf{A}^T\mathbf{K} + \mathbf{K}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{K} - \mathbf{Q} \quad (\text{a.8})$$

$$\mathbf{G} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{K}\mathbf{x} \quad (\text{a.9})$$

として, 制御入力 u が求められ, さらに(a.2)式より $\dot{\delta}_e$ が得られる。ここに, $\mathbf{Q} = \text{diag}(q_1, q_2, q_3)$, $\mathbf{R} = [r]$ である。