

博士学位論文

脳型情報処理アルゴリズムを用いた  
移動ロボットの智能化

平成 22 年 3 月

九州工業大学 大学院 生命体工学研究科

武村 泰範

## 目次

第1章 序論 .....	10
1.1 はじめに .....	11
1.1.1 ロボットの歴史 .....	11
1.1.2 ロボットの知能化 .....	13
1.2 本研究の目的 .....	16
1.3 本論文の構成 .....	16
第2章 知能化の基礎 ～学習～ .....	22
2.1 緒言 .....	23
2.2 階層型ニューラルネットワーク .....	24
2.2.1 階層型ニューラルネットワーク(MLP) .....	24
2.2.2 学習法：誤差逆伝搬学習 .....	25
2.2.3 誤差逆伝播学習の課題 .....	28
2.2.4 誤差逆伝播学習を用いたパターン認識 .....	28
2.3 自己組織化マップ (SOM) .....	29
2.3.1 自己組織化マップの概要と特徴 .....	29
2.3.2 自己組織化マップのアルゴリズム .....	30
2.3.2.1 評価プロセス .....	31
2.3.2.2 競合プロセス .....	31
2.3.2.3 協調プロセス .....	31
2.3.4 適応プロセス .....	32
2.3.3 SOM の学習例 .....	33
2.4 ニューラルガス (NG) .....	34
2.4.1 ニューラルガスの概要と特徴 .....	34
2.4.2 ニューラルガスのアルゴリズム .....	34
2.4.2.1 評価プロセス .....	35
2.4.2.2 競合プロセス .....	35
2.4.2.3 協調プロセス .....	35
2.4.2.4 適応プロセス .....	36
2.4.3 NG による学習例 .....	37
2.5 モジューラ型自己組織化マップ (mnSOM) .....	38

2.5.1	モジュラ型自己組織化マップの概要と特徴 .....	38
2.5.2	モジュラ型自己組織化マップの構造 .....	38
2.5.3	mnSOM の学習アルゴリズム .....	39
2.5.3.1	評価プロセス .....	39
2.5.3.2	競合プロセス .....	40
2.5.3.3	協調プロセス .....	40
2.5.3.4	適応プロセス .....	41
2.5.4	ローカルミニマムの回避方法 .....	42
2.5.5	MLP-mnSOM による学習例 .....	43
<b>第 3 章</b>	<b>知能化の基礎 (ハードウェアの知能化) .....</b>	<b>61</b>
3.1	ハードウェアの知能化 .....	62
3.2	<b>Intelligent Mechanical Design .....</b>	<b>62</b>
3.3	<b>Mechanical Design Principles: “Mecha-telligence Principles” ..</b>	<b>63</b>
3.4	<b>Mechanical Design Process: “Mecha-Telligence Methodology” ...</b>	<b>66</b>
3.5	ハードウェアの知能化例：サッカーロボット .....	67
3.6	ロボカップ .....	68
3.6.1	ロボカップとは .....	68
3.6.2	ロボカップにおける課題 .....	68
3.6.3	ロボカップの競技種類と発展 .....	69
3.7	ロボカップ中型リーグ .....	71
3.8	モジュール構造を用いたプラットフォーム開発 .....	72
3.8.1	全方位移動ロボット “Musashhi” の構成 .....	75
3.8.1.1	Concept of omni-directional mobility .....	75
3.8.1.2	Concept of Safety .....	76
3.8.1.3	Concept of modularity .....	80
3.8.2	全方位移動ロボット”Musashi”のキック機構 .....	83
3.8.3	まとめ .....	84
<b>第 4 章</b>	<b>ソフトウェアの知能化の例 1： 照明環境適応型色認識 アルゴリ</b>	
	<b>ズムの開発 .....</b>	<b>107</b>
4.1	目的 .....	108
4.2	脳型計算処理を用いた照明環境適応型 色認識アルゴリズムの開発	109
4.2.1	研究背景 .....	109

4.2.2 目的.....	111
4.2.3 USB カメラを用いた mnSOM を用いた静的環境における照明環境適応型色認識アルゴリズムの検討 .....	112
4.2.3.1 実験環境.....	112
4.2.3.2 実験手順.....	113
4.2.3.3 実験方法.....	114
4.2.3.4 実験結果.....	115
4.2.3.5 考察.....	117
4.2.4 全方位移動サッカーロボット Musashi に搭載した学習を用いた照明環境適応型色認識アルゴリズムの検討 .....	119
4.2.4.1 ロボカップ中型リーグにおける色抽出.....	119
4.2.4.2 従来の色抽出手法 .....	120
4.2.4.2 提案手法.....	121
4.2.4.2.1 入出力データ .....	121
4.2.4.2.2 自己組織化マップ(SOM)を用いた手法.....	122
4.2.4.2.3 ニューラルガス(NG)を用いた手法 .....	122
4.2.4.2.4 モジューラ型自己組織化マップ(mnSOM) を用いた手法 .....	122
4.2.4.3 実験方法.....	123
4.2.4.4 実験結果.....	126
4.2.4.5 考察.....	127
<b>4.3 まとめ .....</b>	<b>128</b>
<b>第5章 ソフトウェアの知能化例2：自律型水中ロボットの 適応制御システムの開発.....</b>	<b>154</b>
5.1 緒言 .....	155
5.2 代表的な水中ロボット .....	156
5.3 水中ロボットの運動制御 .....	160
5.4. 目的 .....	162
5.5 自律型水中ロボット”Twin-Burger”.....	164
5.6 mnSOM を用いた運動の同定と制御 .....	165
5.6.1 適応制御システム .....	166
5.6.1.1 フォワードモデルマップの作成.....	166
5.6.1.2 コントローラモデルの作成.....	168

5.6.1.3	ロボットへの適応 .....	169
4.6.2	シミュレーション .....	171
4.6.2.1	フォワードモデルマップのシミュレーション 及び考察.....	171
5.6.2.2	コントローラモデルマップのシミュレーション 及び考察 .....	172
5.6.2.4	実機性能実験および考察 .....	173
5.8	まとめ .....	175
<b>第 6 章</b>	<b>考察およびまとめ .....</b>	<b>201</b>
6.1	考察および結論 .....	202
<b>参考文献</b>	<b>.....</b>	<b>206</b>
第 1 章	.....	206
第 2 章	.....	207
第 3 章	.....	208
第 4 章	.....	211
第 5 章	.....	212

## 図表一覽

FIG.1-1 FACTORY AUTOMATION ROBOT (MOTOMAN-SSA2000 YASKAWA)[20] .....	17
FIG.1-2 2-BIPES ROBOT (ASIMO HONDA Co., LTD.) [21] .....	17
FIG.1-3 HUMANOID ROBOT (HRP2 IST)[22] .....	18
FIG.1-4 NEO MECHATORONICS SOCIETY MAP [5] .....	19
FIG.1-5 MARS PATHFINDER “SOJOURNER”[7].....	19
FIG.1-6 MARS EXPLORATION ROVER “OPPORTUNITY” [8] .....	20
FIG.1-7 AUTONOMOUS UNDERWATER VEHICLE “ R2D4 “ [9].....	20
FIG.1-8 MODEL OF INTELLIGENT MODEL.....	21
FIG. 1-9 STRUCTURE OF THIS DOCTOR DISARATION.....	21
FIG.2-1 MODEL OF MULTI LAYER PERCEPTRON.....	44
FIG.2-3 GRAPH OF HYPERBOLIC TANGENT FUNCTION .....	44
FIG.2-2 GRAPH OF SIGMOID FUNCTION .....	44
FIG.2-4RESULT OF BP LEARNING (IRIS) .....	45
FIG.2-5 VECTOR SPACE OF IRIS DATA.....	45
FIG. 2-6 VECTOR SPACE AND FEATURE MAP.....	46
FIG.2-7 ERROR BETWEEN DATA VECTOR AND REFERENCE VECTORS .....	46
FIG. 2-8 ADAPTATION OF REFERENCE VECTORS.....	47
FIG. 2-9 CHANGE OF FEATURE MAP BY BATCH LEARNING .....	48
FIG.2-10 THE RESULT OF BATCH SOM FEATURE MAP.....	49
FIG.2-11 AN UNIFORM DISTRIBUTION LEARNING DATA .....	50
FIG.2-12 THE RESULT OF THE SOM ALGORITHM (AN UNIFORM DISTRIBUTION) .....	50
FIG.2-13 THE RESULT OF THE NG ALGORITHM (AN UNIFORM DISTRIBUTION).....	51
FIG.2-14 THE CIRCLE (A DOUGHNUT) LEARNING DATA .....	51
FIG.2-15 THE RESULT OF SOM ALGORITHM (A DOUGHNUT).....	52
FIG.2-16 THE RESULT OF NG ALGORITHM (A DOUGHNUT) .....	52
FIG.2-17 THE TWO UNIFORM DISTRIBUTION LEARNING DATA.....	53
FIG.2-18 THE RESULT OF SOM ALGORITHM (TWO UNIFORM DISTRIBUTION).....	53
FIG.2-19 THE RESULT OF NG ALGORITHM (TWO UNIFORM DISTRIBUTION) .....	54
FIG.2-20 STRUCTURE OF MNSOM.....	55
FIG.2-22 RESULT OF CUBIC FUNCTION MNSOM.....	56
FIG.2-23 RESULT OF MNSOM FEATURE MAP.....	57

TABLE 2-1 VARIABLE USED EXPLANATION OF SOM ALGORITHM .....	58
TABLE 2-2 ANIMAL DATA VECTORS.....	59
TABLE 2-3 PARAMETERS OF SOM ALGORITHM .....	60
TABLE 2-4 VARIABLE USED EXPLANATION OF NG ALGORITHM .....	60
FIG. 3-1 MECHANICAL DESIGN OF AN AUTONOMOUS MOBILE ROBOT AND RELATED FACTORS [3]	85
FIG. 3-2 NEW APPROACH TO MECHANICAL DESIGN OF AN AUTONOMOUS MOBILE ROBOT .....	85
FIG. 3-3 OVERVIEW OF DESIGN PRINCIPLES OF MECHANICAL DESIGN OF .....	86
FIG. 3-5 SIMULATION LEAGUE.....	87
FIG. 3-6 SMALL SIZE LEAGUE .....	87
FIG. 3-8 4-LEG LEAGUE .....	88
FIG. 3-9 HUMANOID LEAGUE.....	88
FIG. 3-10 RESCUE LEAGUE.....	88
FIG. 3-11 @HOME LEAGUE.....	89
FIG. 3-12 ROBOCUP JR DANCE LEAGUE .....	89
FIG.3-13 ROBOCUP JR. SOCCER LEAGUE .....	89
FIG. 3-14 ROBOCUP JR. RESCUE LEAGUE .....	90
FIG. 3-15 LIMITATION OF A MSL ROBOT SIZE .....	90
FIG. 3-16 THE FIELD OF ROBOCUP MSL (18 x 12 [M]) .....	91
FIG. 3-17 THE SELECTED TYPE OF SOCCER ROBOT [19].....	91
FIG. 3-18 THE HIGH-LEVEL SPECIFICATION LAYERS INCLUDING MAIN-LAYER AND SUB LAYER [19] .....	92
FIG. 3-19 LAYER 1 GENERATED FROM THE SUB-LAYER[19].....	92
FIG. 3-20 LAYER 2 GENERATED FROM THE LAYER 1 [19] .....	93
FIG. 3-21 MONO-SPEC LAYER GENERATED FROM THE LAYER 2 [19].....	93
FIG.3-22 RESULTS OF THE MONO-SPEC LAYER GENERATION AND THE FUNCTION DESIGN PRIORITY [19] .....	94
FIG. 3-23 THE RESULTS OF DESIGN [19] .....	94
FIG. 3-24 THE FIRST VERSION OF OUR ROBOT EQUIPPED WITH ELEVEN SENSORS.....	95
FIG. 3-25 MUSASHI ROBOT PLATFORM .....	95
FIG. 3-31 FLOWCHART OF THE MUSASHI ROBOT .....	100
FIG. 3-32 FLOWCHART OF THE MUSASHI ROBOT: .....	100
FIG.3-33 SIGLE MODULES .....	101
FIGS. 3-34 BATTERY MODULE.....	102

FIG.3-35 MODULES OF “MUSASHI” ROBOT HARDWARE.....	103
FIG.3-36 MODULES OF “MUSASHI” ROBOT .....	104
TABLE 3-1 SPECIFICATION OF A “MUSASHI” ROBOT .....	105
TABLE 3-2 RISK ASSESSMENT OF THE “MUSASHI” ROBOT.....	106
FIG.4-1 TESTING DATA ENVIRONMENT .....	130
FIG.4-2 TESTING DATA .....	130
FIG.4-3 SAMPLE OF COLOR DATA .....	131
FIG. 4-4 BP VISION SYSTEM .....	132
FIG. 4-5 MNSOM’S FUNCTIONAL MODULE IN OUR ALGORITHM .....	132
FIG.4-6 RESULT OF BP LEARNING (SOMG CANVAS) AND ERROR COLOR PICTURE.....	133
FIG.4-7 RESULT OF BP TEST DATA(SOMG CANVAS) AND ERROR PICTURE.....	134
FIG.4-8 RESULT OF MNSOM FEATURE MAP.....	135
FIG.4-9 RESULT OF BP EXPERIMENT II LEARNING DATA ERROR SAMPLE PICTURE.....	136
FIG.4-10 RESULT OF BP EXPERIMENT II TEST DATA .....	136
FIG.4-11 RESULT OF MNSOM FEATURE MAP.....	137
FIG. 4-12 CONVENTIONAL COLOR EXTRACTION ALGORITHM .....	138
FIG. 4-13 THE OMNI-DIRECTIONAL MOBILE SOCCER ROBOT “MUSASHI.....	138
FIG. 4-14 LIGHTING CONDITION OF THE EXPERIMENT: ARROW SHOWS THE ROBOT MOVEMENT .....	139
FIG.4-15 TRANSITION THE UNITS BY SOM ALGORITHM .....	139
FIG. 4-16 TRANSITION OF THE UNIT IN FEATURE MAP BY SOM ALGORITHM .....	140
FIG. 4-17 TRANSITION THE THRESHOLD BY SOM ALGORITHM.....	140
FIG. 4-18 TRANSITION THE UNITS BY NG ALGORITHM .....	141
FIG. 4-19 TRANSITION THE THRESHOLD BY NG ALGORITHM.....	141
FIG. 4-20 TRANSITION THE MODULES BY MNSOM ALGORITHM .....	142
FIG. 4-21 TRANSITION THE MODULES AND FEATURE MAP BY MNSOM ALGORITHM.....	142
FIG.4-22 TRANSITION THE THRESHOLDS BY MNSOM ALGORITHM .....	143
TABLE 4-1 TEST COLOR.....	144
TABLE 4-2 CONDITION OF TEST ENVIRONMENT .....	145
TABLE 4-3 PARAMETERS OF MLP-MNSOM .....	146
TABLE 4-4 RESULT OF BP EXPERIMENT I ABOUT LEARNING DATA.....	146
TABLE 4-5 RESULT OF MLP-MNSOM EXPERIMENT I ABOUT LEARNING DATA.....	147
TABLE 4-6 RESULT OF BP EXPERIMENT I ABOUT NON-LEARNING DATA.....	147

TABLE 4-7 RESULT OF MLP-MNSOM EXPERIMENT I ABOUT NON-LEARNING DATA.....	148
TABLE 4-8 RESULT OF BP EXPERIMENT II ABOUT LEARNING DATA .....	148
TABLE 4-9 RESULT OF MLP-MNSOM EXPERIMENT II ABOUT LEARNING DATA .....	149
TABLE 4-10 RESULT OF BP EXPERIMENT II ABOUT NON-LEARNING DATA .....	149
TABLE 4-11 RESULT OF MLP-MNSOM EXPERIMENT II ABOUT NON-LEARNING DATA .....	150
TABLE 4-12 RESULT OF MLP-MNSOM ABOUT LEARNING DATA (10 BP LEARNING/ EPOCH) .....	150
TABLE 4-13 RESULT OF MLP-MNSOM ABOUT NON LEARNING DATA ( 10 BP LEARNING/ 1 EPOCH) .....	150
TABLE 4-14 VARIABLES USED IN EXPLANATION OF SOM AND NG ALGORITHMS .....	151
TABLE 4-15 VARIABLES USED IN EXPLANATION OF MNSOM ALGORITHM .....	151
TABLE 4-16 ENVIRONMENT OF THE LEARNING DATA .....	152
TABLE 4-17 ENVIRONMENT OF THE NON LEARNING DATA .....	153
TABLE 4-18.....	153
RESULT OF EXPERIMENT I.....	153
FIG.5-1 CLASSIFICATION OF UNDERWATER ROBOTS [5-19 参照] .....	176
FIG.5-2 ALVIN ( <a href="http://www.whoi.edu/marops/vehicles/alvin/index_en8.html">HTTP://WWW.WHOI.EDU/MAROPS/VEHICLES/ALVIN/INDEX_EN8.HTML</a> ).....	176
FIG.5-3 ABE.....	177
FIG. 5-4 ODYSSEY II.....	177
FIG.5-5 REMUS600 .....	177
FIG.5-6 JASON.....	178
FIG.5-7 HYSUB5000 .....	178
FIG.5-8 HEMIRE.....	179
FIG.5-9 しんかい 6500.....	179
FIG.5-10 かいこう(亡失以前) .....	180
FIG.5-11 R-ONE ROBOT.....	180
FIG.5-12 R2D4 .....	181
FIG.5-13 AQUAEXPLORER2000 .....	181
FIG.5-14 TRI-DOG 1 .....	182
FIG. 5-15 AQUA-BOX III.....	182
FIG. 5-16 AUV “ TWIN-BURGER” .....	183
FIG. 5-17 THE NETWORK STRUCTURE OF A FORWARD MODULE AND A CONTROL MODULE .....	185
FIG. 5-18 LEARNING PROCESSES OF AN ADAPTIVE CONTROLLER SYSTEM USING RNN-MNSOM .....	186

FIG. 5-19 A FORWARD MODEL MAP OBTAINED FROM THE TIME SERIES OF LIMIT CYCLE SIMULATION DATA .....	187
FIG. 5-20 FORWARD MODEL MAP EVALUATION IN M-C SPACE CALCULATED BY THE MEANS SQUARE METHOD .....	188
FIG. 5-22 CONTROL EXPERIMENT OF MNSOM USING "TWIN-BURGER" .....	199
TABLE 5-1 SPECIFICATIONS OF TWIN-BURGER.....	200
TABLE 5-2 COEFFICIENT <i>M</i> AND <i>C</i> FOR LIMIT CYCLE MOTION.....	200

# 第 1 章

## 序論

## 1.1 はじめに

### 1.1.1 ロボットの歴史

ロボットは、人が飽きるような単調な繰り返し作業、人の手では困難な高精度・高速な作業、極限環境に代表される人にとって危険な場所での作業を目的として開発されてきた。ロボットの歴史は、産業界と密接に関係しており、ロボットの開発は産業界の発展に大きく貢献している。

ロボットという言葉は、1920年代に作家の Karel Capek によって書かれた戯曲“R. U. R”[1-1]のなかで“労働”を意味するチェコ語を意味する Robotka から作った言葉とされている。1940年代には、Isaac Asimov が著書でロボット工学三原則を発表し、小説や映画にもみられるように人間とロボットの共存について議論の題材となっている[1-2]。

20世紀初頭、ロボットは文学中での架空のものとして取り扱われているが、ロボットが現実に開発されるのは1960年代以降である。ロボットの開発は、人や動物のような形をした巧妙に働く機械を作ってみたいといった願望に端を発し、コンピュータ技術の発達に伴って、技術者にロボットを作る為の強力なツールとして機能し現在のロボットの発展につながっている。コンピュータが様々な外部装置とインターフェースを備えるようになり、コンピュータの高い計算処理能力のロボットに新境地が開いた。1962年、マサチューセッツ工科大学のリンカーン研究所で H. Ernst は、放射性同位体元素を取り扱う為の開発されていたマニピュレータをコンピュータで動かす研究を行った[1-3]。1963年には、L. Roberts らは、テレビカメラをコンピュータにつないで映像データを読み込み、写っている物体を解析・認識する研究を発表した。

1970-1980年代は、生産システムの中で使われるロボットが急速に進歩した時期である。産業用ロボットは、部品やツールを移動させ仕事をこなす。その為、部品をつかむハンドや塗装のスプレー、溶接ガンなどマニピュレータのエンドエフェクタを変更させ、その位置と姿勢を制御する為の6自由度マニピュレータの制御技術が必要となった。産業用ロボットは、アメリカの AMF 社や日本の安川電機等において開発され、第一次ロボットブームが巻き起こる(Fig. 1-1)。また、1970年代はマイクロコンピュータが開発され、コンピュータの集積化が進み産業用ロボットに組み込む事が可能となった為、小型化が進み応用分野が

広がった。この時期にファクトリーオートメーション化(FA 化)が進み産業界へのロボットの導入が一気に加速した。並行してこの時期に視覚によるセンシング技術が発達した。しかし、この時期以降、産業用ロボットの盛衰とは裏腹に、ロボットの研究活動は基礎技術の深化に向かい大学での研究と産業応用とが分化した。

ロボットは、未来を先取りつつ人間が作り出してきた人工物である。外観が人や動物に似ていても、ただ動くだけではロボットと呼ぶわけにはいかない。ロボットには、機械要素によって構成された身体とコンピュータ技術の応用によって生まれる**知能**が必要であり、自分を取り巻く状況を認識し、適切に行動することが要求される。開発されたロボットは、ユーザや開発者である人間とのインタラクションを通じて動作し、人間に有益でなければならない。ロボットは、様々な要素技術が必要であり、要素技術が統合された統合システムである。

ロボットの技術開発要素は、大きく分けてアームやハンドや移動機能のようなモーション系、視覚やセンサを用いた触覚・聴覚など五感に変わるセンシング系、計画立案や記憶、対話などのインテリジェンス系に分類され、ロボットとして機能するにはこれらの統合が大きな課題である。

1990年代に入り、要素技術を複合して実現した自律型の移動ロボットの開発が行われている。例えば、二足歩行が可能となった”ASIMO”(Fig.1-2)や”HRP-2”(Fig.1-3)などが開発され、柔らかい動作や人間に近い動作を行う事が可能となっている。それ以外の分野としても、人間の作業が困難な水中で動く海中ロボットや宇宙ロボット、またエンターテインメント性や癒しを持ったロボットなども現在では活躍している[1-4]。

現在のロボットの開発傾向を示す経済産業省が提案する10年後のネオ・メカトロニクス社会の展開図(fig.1-4)[1-5]では、居住、商業、娯楽、オフィス、産業、交通、医療などの様々な分野にロボットが応用されている様子が描かれている。このようにロボットは、経済界においても期待されている能力が多岐にわたり、様々な分野での活躍が期待されている。現在、ロボットは主に工場内や人間による直接的な作業が困難な極限環境において活躍している。例えば、宇宙では、NASAによる火星探査機[1-6]として1996年に発射された Mars Pathfinder “Sojourner”(Fig.1-5)[1-7]や2003年に打ち上げられた Mars Exploration Rover “Opportunity”(Fig.1-6)[1-8]などがある。海底では、東京大学生産技術研究所によつ

て開発された”r2D4”(Fig.1-7)[1-9]によって海底火山の調査が行われた[1-10].

このようにロボットの歴史を紐解いていくとある今後のロボット発展を考える上で重要な言葉が浮き上がる. それらは, 自律性: 環境認識による行動選択の的確さ, 汎用性: 様々な作業に適用可能か, 知能性: 学習により作業精度が向上するか, 身体性: 環境/作業に適した形状をしているか, であり, 開発コストを考慮してロボット開発が進められる. ロボットが高度な作業をする為には, ロボットは, 考える, 学習する, 自分で状況を判断して行動を創発し, 自律的に行動を行う必要がある. そこで, 本稿では, ロボットの知能化を研究のテーマとした.

### 1.1.2 ロボットの知能化

知能は, 単純な定義でとらえるには複雑すぎる概念である. 一般の人々だけでなく, 科学者にとってさえこの言葉はあまりにも幅広い意味を持つ. ペンギン社の心理学辞書(the Penguin Dictionary of Psychology)によると, 知能の項目には, “心理学では知能についてのいくつかの概念は注目されているが, 十分な解明は進んでいない”といわれており, 知能の定義することの難しさを語っている[1-11]. 1921年には, 教育心理学ジャーナル(the Journal of Educational Psychology)において, 14人の研究者による知能の定義を行った[1-12]. 14人から届いた返答は全く異なる物であった. いくつかの例を示すと, 抽象的に思考する能力(L. M. Terman), 環境への順応を学習する事またその能力(S. S. Colvin), 新しい状況に適切に対処する能力(R. Pintner), 多様な刺激を統合し, 何らかの統一的な振る舞いを見いだす生物学的メカニズム(J. Peterson), 能力を獲得する為の能力(H. Woodrow)などである. このように各人の定義は異なりながらも, 重要なポイントを示している. 例えば, Termanは, 抽象的思考能力について言及しているのに対して, Petersonは, 生物学的メカニズムについて述べている. この様などのようなアプローチを持って定義していくかによって知能は大きく意味が変化する. このように知能化は, 各々のアプローチの仕方によって変化するものであり一概に決められるものではない. そこで, 自分なりのロボットの知能化へのアプローチを考えてみる.

ロボットを知能化するためのアプローチを考えると, 前節に示す様に今後の発展するためのキーワードからヒントを得る事が出来る. それらは, 自律性:

環境認識による行動選択の的確さ，汎用性：様々な作業に適用可能か，身体性：環境／作業に適した形状をしているかである．これらのキーワードを考えると，ロボットが智能化する為には，ソフトウェアを開発する際に必要な自律性やハードウェアを開発するときに必要な汎用性や身体性といった部分を考慮しながら智能化していく必要がある．そこでまず，ソフトウェア的な智能化について考える．

ロボットのソフトウェア的な智能化に関して考えると，Phil Johnson-Laird の著書“計算機と心(The Computer and the Mind) [1-18]”からヒントを得る事が出来る．Johnson-Laird は，知能を持たせる事の主なタスクを以下のように仮定する事から始めている：

- ・外界を知覚すること
- ・学習する事，記憶する事，行動を制御すること
- ・思考する事，新しいアイデアを創り出すこと
- ・他者とのコミュニケーションを制御すること
- ・感情，意志，自意識に関する経験を作ること

Johnson-Laird の著書では，これらのタスクのおののおのについて計算論的立場に基づいた説明がなされている．彼の主張によると，知能の理論は，コンピュータによってプログラムしてモデル化することが可能であると言われている．そこで，智能化を行う為にエージェントをモデル化すると Fig.1-8 に示すようなモデルが用いられる．まず，外界を近くする為のセンシング技術が必要となる．このセンシング技術を学習や記憶などの機能を持たせることで智能化をする．また，制御部分についても学習の技術を取り入れることで高度な制御を実現する[19]．

例えば，このように学習や記憶，経験といったものを利用して智能化する為には，生物学的エージェントの行動をモデリングする手法がある．このようなモデリングの一例として，コオロギの騒音性に関する振る舞いを模擬する為に自律ロボットが開発された[1-13,14]．このロボットを用いてロボットからコオロギの振る舞いについての仮説を作り上げ，本物のコオロギの行動と比較し，検証した．また，逆に生物学的なアプローチをロボットの智能化へ応用させるアプローチも行われてきた．Lambrinos らは，砂漠アリ(Cata-Glyphis)と同じ原理に従って蟻のえさ追い行動におけるナビゲーションを模倣したロボットを開発

した[1-15].また, Ferrell らは, Cruse によって提唱された昆虫の歩行に関する原理を応用する事により, 歩行ロボットを開発した[1-16,17].

このように生物の仕組みを用いることは重要なアプローチ手段である. これまでの研究として, 昆虫以外にも人間の脳の処理能力を応用したアプローチが数多くなされている.これは, ロボットの自律化を行う上でも有用なアプローチのひとつとして取り扱われてきた. そこで本稿では, **脳型計算処理アルゴリズム**を用いることでソフトウェアとしての知能化を実現する事が出来るのではないかと仮説をたてた.

ここまでは, ソフトウェア的なアプローチからロボットの知能化を考えてきたが, ハードウェア的なアプローチからのロボットの知能化を考える. ロボットが行動する際に, 高度な知能を持って行動する為には, プラットフォーム自体の性能も大きく関わってくる. つまり, ハードウェアの設計上の段階に置いても, 操作性, 拡張性, 信頼性, 安全性などの面を考慮しなければ, ソフトウェアへの比重が大きくなる. ハードウェアの性能は, ソフトの複雑な設計とトレードオフな関係が成り立つと考えられる. そこで我々は, ハードウェアの知能化された設計が必要であると考えた. そこでハードウェアの知能化を以下のように定義する. **ハードウェアの知能化とはロボットが設計段階で性能を最大限に活かせるよう設計する指針を示す.**

本研究では, ソフトウェア・ハードウェア双方から知能化とする事でロボットの知能化を達成する事ができるのではないかと仮説をたてた.

## 1.2 本研究の目的

本研究の目的は、ロボットの知能化を行う事を目的とし、ハードウェアとソフトウェアの両面からの知能化を提案する。具体的な例として、ハードウェアの知能化は、自律型移動サッカーロボットの開発を通じて述べる。ソフトウェアの知能化には、前節で取り上げた脳を模倣した脳型情報処理アルゴリズムを用いる。具体的な応用例として、センシング技術の向上を目的とし、学習を用いた照明環境認識型色認識アルゴリズムの開発を行った。また、水中ロボットの自律行動制御を行う為に、ロボットの制御部分に関して、ロボットのダイナミクスを脳型情報処理アルゴリズムによって推定し、コントローラを変更するアルゴリズムを開発した。

## 1.3 本論文の構成

本論文の構成は、Fig.1-9 のように構成されている。1章においてロボットの開発動向及びロボットの知能化に関して述べる。2章では、ロボットのソフトウェアの知能化のツールとして使用する脳型情報処理アルゴリズムについて述べる。具体的には、パターン認識手法として用いられている自己組織化マップ(SOM: Self-Organizing Map), ニューラルガス(NG: Neural Gas), モジュール型自己組織化マップ(mnSOM: modular network SOM)について扱う。3章では、ハードウェアの知能化に関しての基礎知識として、アミールによって提案されたロボットの設計手法に関して述べ、ハードウェアの知能化として全方位移動サッカーロボットのハードウェア開発について述べる。4章では、ソフトウェアの知能化のアプリケーションとして学習を用いた照明環境適応型色認識アルゴリズムの開発について述べる。5章では、水中ロボットの適応制御システムの開発について述べる。最後に6章は、考察およびまとめを述べる。



Fig.1-1 Factory Automation Robot (Motoman-SSA2000 YASKAWA)[20]

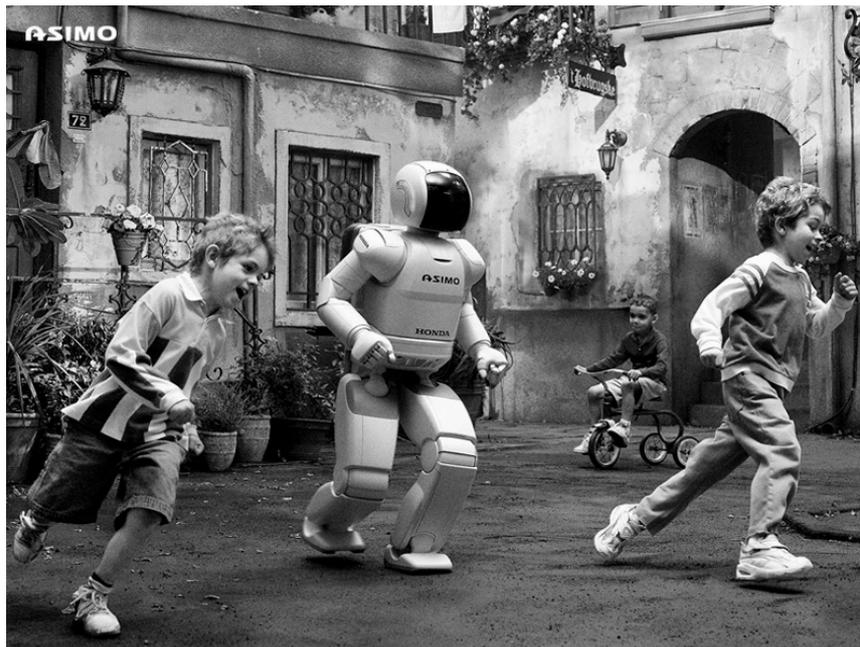


Fig.1-2 2-bipes robot (ASIMO HONDA Co., Ltd.) [21]



Fig.1-3 Humanoid Robot (HRP2 IST)[22]





Fig.1-6 Mars Exploration Rover “Opportunity” [8]

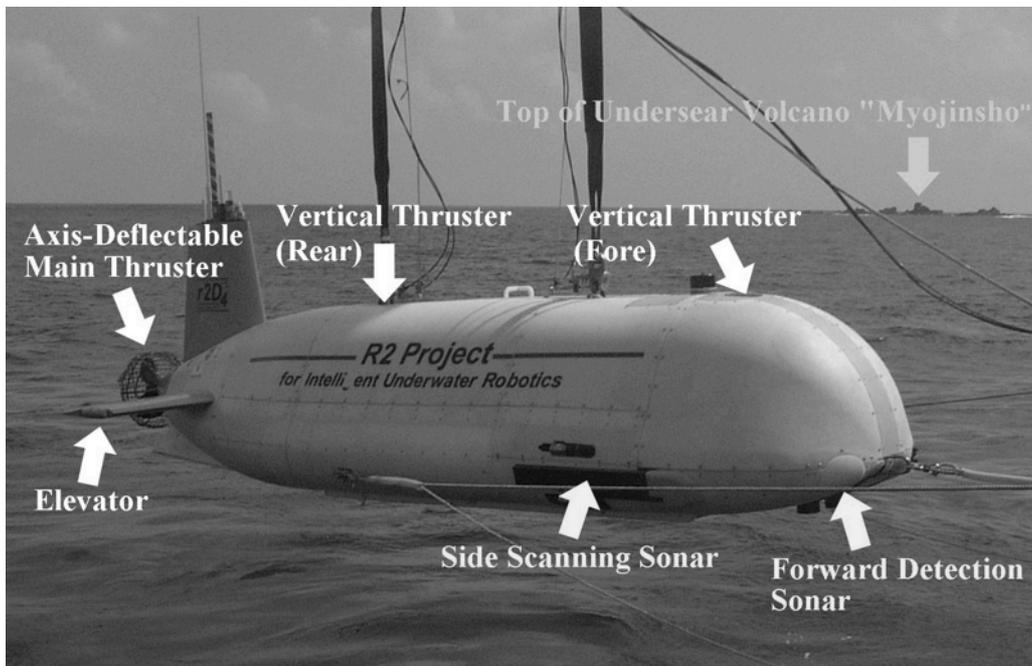


Fig.1-7 Autonomous Underwater Vehicle “r2D4” [9]

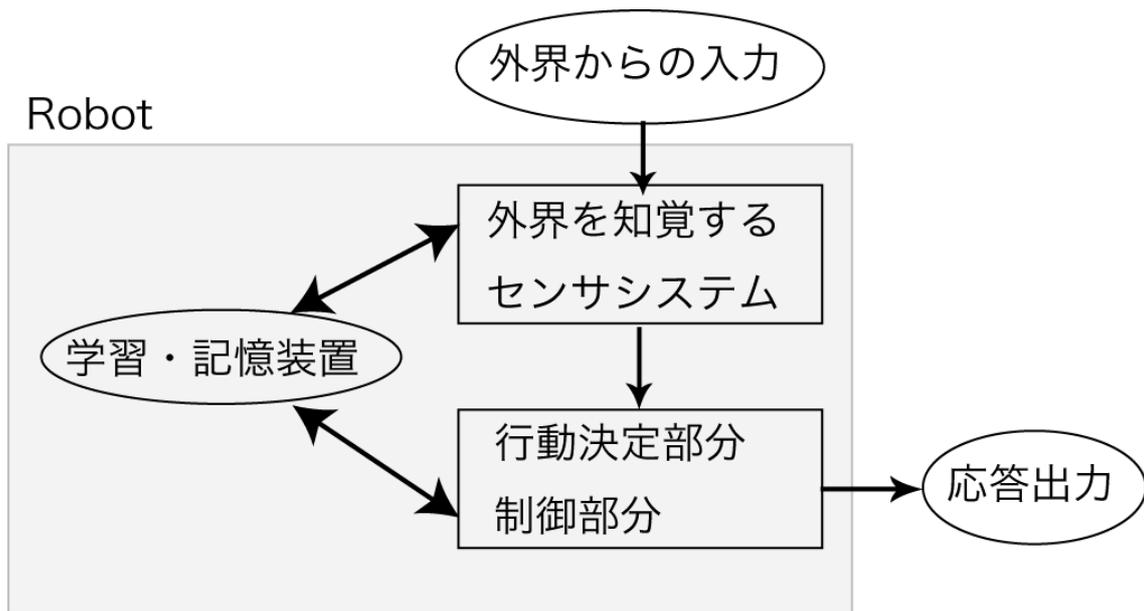


Fig.1-8 Model of intelligent model

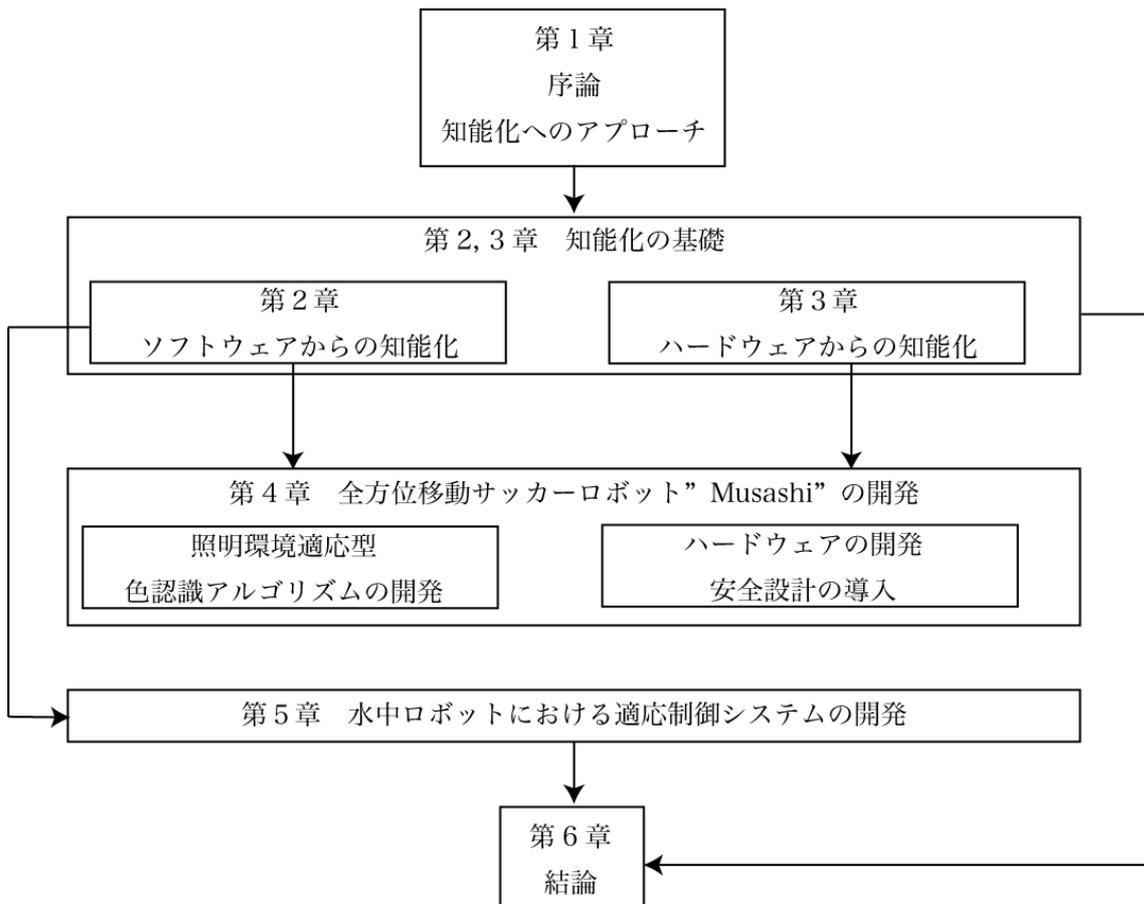


Fig. 1-9 Structure of this doctor disaration

第 2 章  
知能化の基礎  
～学習～

## 2.1 緒言

本章では、自律型移動ロボットのソフトウェアからの知能化アプローチについて議論するため、ニューラルネットワークに代表される学習機能を有するアルゴリズムを取り上げる。ニューラルネットワークの特徴として、ニューラルネットワークは様々なシステムに対して寛容であり、学習を行う事で非線形的な信号に対しても適応可能であるようにロバスタである。ニューラルネットワークは、生体の脳における神経生理学の知見を用いて脳の処理機能を模倣し情報処理を実現している。神経系は、並列に活動して情報を処理する非常に多数のニューロンから構成されており、神経生理学の重要な知見の一つである神経系の並列演算処理能力を実現している。並列性をもつ重要な利点は、個々のニューロンの情報処理速度が遅くても、大量の情報処理が可能なことであり、ロバスト性に寄与している。例えば、ネットワークの構成要素間には 100 万の結合がある場合、どの結合を強化するべきかを解析的もしくは試行錯誤的に見いだすことは非常に困難であるため、学習アルゴリズムが重要となる[2-19]。経験から学習し、結合の強さを自動調節できる事は、1章で示した知能化の定義に合致する。本論文では、ニューラルネットワーク（脳型情報処理アルゴリズム）を採用する事で知能化を目指す。

ニューラルネットワークは、ロボットの制御や画像解析、医療など様々な分野でソフトコンピューティング技術のひとつとして応用されてきた。ニューラルネットワークにはパーセプトロン、リカレント型ニューラルネットワーク、ホップフィールド、自己組織化マップなど様々な種類のものが存在しており、それぞれの特徴を活かした課題が進められている。例えば、リカレント型ニューラルネットワークでは、本研究室で使用されている段差乗り越えロボットの制御システム[1-5]や、自己組織化マップを用いた水中ロボットの行動制御[2-6]などがあげられる。

本章では、階層型ニューラルネットワーク（バックプロパゲーション）、自己組織化マップ、ニューラルガス、モジュール型自己組織化マップについて述べる。

## 2.2 階層型ニューラルネットワーク

### 2.2.1 階層型ニューラルネットワーク (MLP)

階層型ニューラルネットワークは、入出力間の写像関係を学習して構築され、信号が常に入力層から出力層方向に流れる基本的かつ解析の容易な脳型情報処理アルゴリズムである。入力データと出力目標データの対からなる訓練データを多数組与え、ネットワークがこれらの入出力関係を再現するよう結合重みを調整するのがニューラルネットワークである[2-7]。

Fig. 2-1 は典型的な3層の階層型ニューラルネットワークを示す。下から入力層、隠れ層（中間層）、出力層、各層に属するユニットを入力ユニット、隠れ（中間）ユニット、出力ユニットと呼ぶ。入力層と隠れ層間、および隠れ層と出力間は密結合であることが多いが、RBF（radial basis function）のように隠れユニットが一部の入力ユニットに結合するネットワーク構成も提案されている。入力層と出力層はネットワークがその外界とのインターフェースとなるため、写像対象に応じてユニット数が定まる。隠れ層は外界とは直接入出力がなく、隠れユニットの数を定める明確な基準はない。

学習アルゴリズムについて以下に示す。まず、ネットワークに訓練パターン（学習データ）を入力層に与え、通常は正規化の後、中間層ユニットへ伝達する。中間層の各ユニットでは(2.1)式を用いて活性度を計算する。その活性度をシグモイド状の関数を用いて変換し、ユニットの出力とする(Fig.2-2, 2.3 式、またはハイパボリックタンジェント(Fig.2-3)(2.4 式))。得られた出力を後段の中間層域または出力層に伝搬し、順次、各層におけるユニット  $i$  の活性値  $x_i$  を計算する。この活性値を生体自身のニューロンでは膜電位または内部ポテンシャルと呼ぶ。また、2.1 式における  $b_i$  は、バイアス値である。

$$x_i = \sum_{i=1}^n w_{ij} y_i + b_i \quad (2.1)$$

$$y_i = f(x_i) \quad (2.2)$$

$$y = \frac{1}{1+e^{-x}} \quad (2.3)$$

$$y = \tanh(x) \quad (2.4)$$

## 2.2.2 学習法：誤差逆伝搬学習

誤差逆伝搬学習は、ネットワークの出力値と目標値（教示データ）の誤差を出力層から入力層に向け、逆方向に伝搬させて結合加重を調整する学習法である。一般的には、(2.5)式に示される出力値の二乗誤差和  $E$  を減少させるように結合加重を調整する学習法である(2.5 式)。

$$E = \sum_k E_k = \frac{1}{2} \sum_k \sum_i (y_i^{(k)} - d_i^{(k)})^2 \quad (2.5)$$

$E_k$  は、 $k$  番目の訓練パターンに対する出力の二乗誤差を表しており、 $y_i^{(k)}$  は、訓練パターン  $k$  に対する出力ユニット  $i$  の出力値を示し、 $d_i^{(k)}$  を出力目標値である。最急降下法を用い、 $E$  を結合加重の関数とみなして  $E$  が減少する方向へ結合加重を微少量移動させる。

学習の方法には大別して逐次学習とバッチ学習の 2 種類存在する。逐次学習は、訓練パターン毎に学習を行う、つまり各  $E_k$  に対して結合加重を修正する。バッチ学習は、すべての訓練パターンが提示された後に  $E$  を用いて結合重みを修正する方法である。両者の違いは結合加重の更新間隔にあるので、ここでは逐次学習について述べる。

逐次学習の結合重み修正

結合重み  $w_{ij}$  に関する評価  $E_k$  の偏微分を計算する。

$$\Delta w_{ij} = -\eta \frac{\partial E_k}{\partial w_{ij}} \quad (2.6)$$

$\Delta w_{ij}$ : 結合重みの修正量       $\eta$ : 偏微分にかかる係数で学習率を示す

(2.6) 式にマイナス符号があるのは、 $E_k$  が小さくなる方向に結合重みを修正するためである。

(2.6) 式の偏微分を合成関数の微分とすると

$$\frac{\partial E_k}{\partial w_{ij}} = \frac{\partial E_k}{\partial x_i^{(k)}} \frac{\partial x_i^{(k)}}{\partial w_{ij}} \quad (2.7)$$

$x_i^{(k)}$ : 訓練パターン  $k$  に対するユニット  $i$  の活性値（入力の負荷和）

(2.1)式を参照

よって、(2.7) 式右辺第 2 項は(2.1)式より

$$\frac{\partial x_i^{(k)}}{\partial w_{ij}} = y_j^{(k)} \quad (2.8)$$

$y_j^{(k)}$  :  $x_i^{(k)}$  の一つ下位層ユニット  $j$  の出力値である.

(2.6)式の右辺第 1 項に負符号をつけたものを

$$\delta_i^{(k)} = -\frac{E_k}{\partial x_i^{(k)}} \quad (2.9)$$

$\delta_i^{(k)}$  : ユニット  $i$  の誤差信号

出力自乗誤差和のユニット  $i$  の活性値に対する感度

(2.9)式を変更すると

$$\delta_i^{(k)} = \frac{\partial E_k}{\partial y_i^{(k)}} \frac{\partial y_i^{(k)}}{\partial x_i^{(k)}} \quad (2.10)$$

$y_i^{(k)}$  : 活性値  $x_i^{(k)}$  に対応する出力値

ここで(2.9)式の右辺第 2 項は、

$$\frac{\partial y_i^{(k)}}{\partial x_i^{(k)}} = f'(x_i^{(k)}) \quad (2.11)$$

(2.11)式の右辺第 1 項は、出力値  $y_i^{(k)}$  が

- (a) 出力ユニットの出力値の場合
- (b) 隠れユニットの出力値の場合

によって場合わけが必要となる.

(a) 出力ユニットの場合

$$-\frac{\partial E_k}{\partial y_i^{(k)}} = -\frac{1}{2} \frac{\partial \sum_j (y_j^{(k)} - d_j^{(k)})^2}{\partial y_i^{(k)}} = d_i^{(k)} - y_i^{(k)} \quad (2.12)$$

(2.10)(2.11)(2.12)式より

$$\delta_i^{(k)} = (d_i^{(k)} - y_i^{(k)}) f'(x_i^{(k)}) \quad (2.13)$$

となる.

(b) 隠れユニットの出力の場合

一つ上位の階層のユニット  $j$  の活性度  $x_j^{(k)}$  および、誤差信号  $\delta_j^{(k)}$  がすでに計算済みとすると

$$-\frac{\partial E_k}{\partial y_i^{(k)}} = \sum_j \frac{\partial E_k}{\partial x_j^{(k)}} \frac{\partial x_j^{(k)}}{\partial y_i^{(k)}} \quad (2.14)$$

ここで (2.14) 式の右辺第 1 項は(2.9)式より  $\delta_j^{(k)}$  であり右辺第 2 項は(2.1)式より

$$(2.15)$$

よって、(2.10)(2.11)(2.15)式より

$$\delta_i^{(k)} = f'(x_i^{(k)}) \sum_j \delta_j^{(k)} w_{ji} \quad (2.16)$$

よって、上位階層の誤差信号  $\delta_j^{(k)}$  を結合加重  $w_{ji}$  を介して、当該階層に逆伝播されてえられた  $\sum_j \delta_j^{(k)} w_{ji}$  に、当該階層の  $f'(x_i^{(k)})$  をかけることによって隠れユニットの誤差信号  $\delta_i^{(k)}$  が再帰的に決まる。

以上より、結合加重の修正量は次式で与えられる

$$\Delta w_{ij} = \eta \delta_i^{(k)} y_j^{(k)} \quad (2.17)$$

このようにして学習は出力層から入力層の方の重みを修正していく。

また、振動を制御するためにモーメンタム項を利用する.学習の終わりでは 2.17 式の代わりに(2.18)式を用いる

$$\Delta w_{ij} = \eta \delta_i^{(k)} y_j^{(k)} + \alpha \Delta w_{ij-1} \quad (2.18)$$

$\alpha$  はモーメンタム係数で ( $0 < \alpha < 1$ ) となっており慣性の働きを起こす.また、結合重みの修正量が入ることで、狭谷状の誤差局面の場合、振動を制御し学習速度を速める[2-7].

### 2.2.3 誤差逆伝播学習の課題

誤差逆伝播学習アルゴリズムにはいくつかの課題が存在する。極小値に達する前に局所最小に陥って学習が進まなくなるローカルミニマム問題や、中間層ユニットの数を選択する際に経験的に決定する事が多く最適なユニット数が不明確、論理関数学習機械よりは非線形関数再現機とみるべきであるという点、複雑な論理演算は学習困難であるなどの問題がある。

これらの問題に関してはいくつかの解決方法があり、中間層ユニットはなるべく多く用意しておき徐々に減らす方法によって最適ユニット数を設定する手法が提案されている。石川によって提唱された忘却アルゴリズム[2-8]や安井によって提唱された CSDF アルゴリズムの利用等がある[2-10]。

初期状態で中間ユニットを多めにすることにより、ローカルミニマム回避に寄与する。

### 2.2.4 誤差逆伝播学習を用いたパターン認識

誤差逆伝播学習は、パターン認識の分野において有効な手段である。ここでは逐次型の誤差逆伝播学習を用いパターン認識を行った結果を示す。

パターン識別の例として、植物のアヤメの分類を行った。アヤメの種類を A,B,C の3クラスに分ける問題とし、特徴量はがくの長さ・幅、花卉の長さ・幅とした。これらの特徴量を[-1,+1]までの範囲に正規化を行い伝達関数には  $\tanh(x)$  を使用して実験を行った。50000回学習した実験の結果を Fig.2-4 に示す。3列の構成で各列が1列目をクラス A (赤), 2列目をクラス B (青), 3列目をクラス C (緑) とし、認識不能な場合は黒で表示している。実際に分類する場合事前にわかっているクラス構成を Fig.2-5 に示す。このようにクラス B とクラス C はある程度同じような特徴量を持っているので誤認識を起こすが、高い確率でクラス分けに成功していることがわかる。

## 2.3 自己組織化マップ (SOM)

### 2.3.1 自己組織化マップの概要と特徴

自己組織化マップ(Self-Organizing Map:SOM)は, T.Kohonen によって考案されたニューラルネットワークである[2-11,12,13].SOM は与えられたデータから自己組織的にネットワークを構築するクラスタリング手法の一つである.

前節で説明した誤差逆伝播学習は教師あり学習であるが, SOM は教師なし学習であり, 入力データから競合層のユニットを修正する学習法である.高次元ベクトル空間に存在するデータをトポロジー保存しながら低次元に写像できる特徴を持っている.よって, SOM はデータマイニング, データ分類, データの可視化などにおいて有力な手法である.現在ではプロセス解析, 制御, さらにはウェブ検索システム, 生命体工学, 経営などの実社会において重要な分野に応用されている[2-14]

基本的な SOM には入出力関係はなく, 学習によって形成する競合層が大きな意味を持ち, これを特徴マップと呼ぶ.この特徴マップは高次元ベクトルで表現された入力データを低次元化したものである(Fig.2-6).特徴マップ上では, 入力空間(高次元のベクトル空間)において距離が大きいデータ群は特徴マップにおいても遠くに配置され, 入力空間において距離が近い場合は, 特徴マップにおいても近くに配置される.高次元ベクトルの位置関係では直感的にわからなかった関係が特徴マップにおいて視覚的に理解することが出来る [2-15].

### 2.3.2 自己組織化マップのアルゴリズム

SOMの学習アルゴリズムを以下に示す. 競合層には, 参照ベクトル (Reference vector) を持つユニット(Unit)を格子状に配列されており, 学習により競合層ユニットを修正していく. アルゴリズムの違いから Online-learning SOM と Batch-learning SOM の2種類の学習方法が存在する. アルゴリズムの説明に際し使用する変数を Table 2-1 に示す.

学習アルゴリズムは評価プロセス, 競合プロセス, 協調プロセス, 適応プロセスによって構成されている. ここでは, Batch型SOMの学習方法について述べる. なお, 下付きインデックスは, 入力データに関する添え字を示しており, 上付きインデックスには, ユニットの番号を示している.

### 2.3.2.1 評価プロセス

評価プロセスでは、全クラスのデータ  $x_i$  と各ユニットの参照ベクトル  $w^k$  との距離  $E_i^k$  を計算する(Fig.2-7).

$$E_i^k = \|w^k - x_i\|^2 \quad (2.19)$$

### 2.3.2.2 競合プロセス

評価プロセスで求められた距離  $E_i^k$  が全ユニットの中で最小となるユニットを  $i$  番目のクラスの勝者ユニット(BMU: Best Matching Unit)  $k^*$  とする(2.20 式) (Fig. 2-7)

$$k^* = \arg_k \min E_i^k \quad (2.20)$$

### 2.3.2.3 協調プロセス

近傍関数  $\phi_i^k$  を用いて勝者ユニット近傍のユニットの修正量を分配する.  $i$  番目のクラスに対する各ユニットの学習率を 2.21,2.22 式により決定する. 近傍関数は BMU に対しては 1.0, 距離に応じて 0 に漸近する関数である. ここではガウス関数状の関数を導入する. また, 学習開始時の近傍係数は裾野が広いガウス関数形状とし, 学習が進むにつれて裾野が狭くなる (2.23 式).

$$\phi_i^k = \exp\left(-\frac{d(k, k_i^*)^2}{2\sigma^2}\right) \quad (2.21)$$

$$\psi_i^k = \frac{\phi_i^k}{\sum_i \phi_i^k} \quad (2.22)$$

$$\sigma = \sigma_{\min} + (\sigma_{\max} - \sigma_{\min}) \exp\left(-\frac{t}{\tau}\right) \quad (2.23)$$

ここで $\sigma_{\min}$ は最小の近傍半径で、 $\sigma_{\max}$ は、最大の近傍半径である。 $t$ は学習回数  
を表している。近傍半径によって、BMUからどの範囲まで学習率を分配するかを  
決定する。学習率が決定され、学習率を正規化する事で均一な学習率  $[0, 1]$ が決  
定される(2.22 式)。学習率は近傍半径が小さくするにつれて、小さくなり最終的  
には収束する。

### 3.2.4 適応プロセス

協調プロセスによって求められた学習率 $\psi_i^k$ に従って、各ユニットの参照ベク  
トル $w^k$ を更新する (2.24 式) (Fig.2-8) .

$$w^k(t+1) = \sum_i \psi_i^k x_i \quad (2.24)$$

以上の4つのプロセスを繰り返すことで特徴マップを生成し、データ間の位置  
関係を可視化する[2-16].

### 2.3.3 SOM の学習例

本節では、Batch 型 SOM による学習結果について述べる。学習は動物データに対して行った。動物データはクラス数 17、次元数（データベクトル数）21 である。動物の種類をクラス、特徴量を次元と呼ぶ。特徴量の値の範囲は 0.0～1.0 までの値とした。本実験に使用した動物データの表を Table 2-2 に示す。

SOM の学習アルゴリズムにおける各種パラメータの値を Table 2-3 に示す。Table 2-2 に示したデータ列を使用し、各種パラメータを Table 2-3 のように調節すると Fig.2-9 に示される学習結果が得られた。SOM の特徴マップの格子内の色は、隣り合うユニット間の距離を色で表示しており、青→緑→赤の順で近くから遠くなる様に色づけしている。学習回数を 1 回、10 回、50 回、100 回、200 回、500 回と設定して学習を行った。Fig.2-9 から、200 回以降は変化がなく、近傍半径も小さくなったので学習は収束したと考えられる。

Fig. 2-10 に Batch 型 SOM による結果から考察できるクラスタリング結果を示す。Batch 型 SOM の学習では、まず 2 つのグループに分かれており、鳥類か哺乳類と分類されている。また、最終結果（Fig.2-10）を見て分かるように最終的に哺乳類や鳥類の中でも肉食・草食に分類され、大型中型にも分類されている。鳥類と哺乳類には大きなデータベクトルに差があることも中心付近のマップが赤く表示されていることが推測される。

このように特徴マップを見ることによって多次元のデータベクトルを用意に位置関係やクラス分けを行うことが出来ることを確認した。また、学習は BP 学習に比べかなり早い速度で収束する。実験を何度か行ったが結果には再現性があり、計算時間も短いので安定性が高いことがわかる。

## 2.4 ニューラルガス (NG)

### 2.4.1 ニューラルガスの概要と特徴

ニューラルガス(NG)は, T.Martinetz によって考案されたニューラルネットワークである[2-17,18]. NG も入力データから自己組織的にネットワークを構築するアルゴリズムであり SOM を拡張したアルゴリズムとみなすことができる.

NG の特徴は, SOM と同じ教師なし学習である. SOM は高次元ベクトル空間に存在するデータをトポロジー保存しながら低次元に写像できることが特徴であることに対して, NG は, 学習関数の配分が順位学習となっている為, 各ユニットが独立して移動することが出来ることが特徴である.よって, ユニットが周りの近傍との関係を持たないため収束が速い.

この特徴よりデータマイニング, データ分類, データの可視化などにおいて有力な手法のひとつとされている.

### 2.4.2 ニューラルガスのアルゴリズム

NG は, SOM と同様に参照ベクトル (Reference vector) を持つユニット(Unit)が存在している.アルゴリズムの説明に際して使用する変数を Table 2-4 に示す.

NG の学習アルゴリズムは SOM と同様に以下の評価プロセス, 競合プロセス, 協調プロセス, 適応プロセスによって構成されている.以下に各プロセスについて述べる.なお, 下付きインデックスは, 入力データに関する添え字を示しており, 上付きインデックスに関しては, ユニットに関数添え字を示している.

### 2.4.2.1 評価プロセス

評価プロセスでは、全クラスのデータ  $x_i$  と各ユニットの参照ベクトル  $w^k$  との距離  $E_i^k$  を計算する.

$$E_i^k = \|w^k - x_i\|^2 \quad (2.25)$$

### 2.4.2.2 競合プロセス

評価プロセスで求められた距離  $E_i^k$  が全ユニットの中で最小となる参照ベクトルを  $i$  番目のクラスの BMU を  $k^*$  とする(2.26 式)

評価プロセスによって計算したユニット間の距離を 2.27 式と示すと、2.28 式によって、距離の大きさによって並び替えることが出来る. このとき、Index は  $E$  の距離によってソートされたランキングの番号を示す. なお、添字の  $S$  は、ソートを行う事を意味する.

$$k^* = \arg_k \min E_i^k \quad (2.26)$$

$$E^k = (E_1^k, E_2^k, \dots, E_j^k) \quad (2.27)$$

$$i = \text{index}^S E^k(j) \quad (2.28)$$

### 2.4.2.3 協調プロセス

近傍関数  $\phi_i^k$  を計算して用いることで、各ユニットの  $i$  番目のクラスに対する学習率を分配する (2.29,2.30 式). SOM と同様、近傍関数は BMU に対しては 1.0, 距離に応じて 0 に漸近する関数である (2.31 式).

$$\phi_i^{S E^k(j)} = \exp\left(-\frac{j}{2\sigma^2}\right) U \quad (2.29)$$

$$\psi_i^k = \frac{\phi_i^k}{\sum_i \phi_i^k} \quad (2.30)$$

$$\sigma = \sigma_{\min} + (\sigma_{\max} - \sigma_{\min}) \exp\left(-\frac{t}{\tau}\right) \quad (2.31)$$

ここで $\sigma_{\min}$ は最小の近傍半径で、 $\sigma_{\max}$ は、最大の近傍半径である。 $t$ は、学習回数を表している。この近傍半径を使うことによって、学習を BMU からどの範囲まで学習するかを決定する。これによって、学習率が決定され、それらを正規化する形で学習率が決定される。学習率は近傍半径が小さくするにつれて、小さくなり最終的には収束する。

#### 2.4.2.4 適応プロセス

協調プロセスによって求められた学習率 $\psi_i^k$ に従って、各ユニットの参照ベクトル $w^k$ を更新する (2.32 式)。

$$w^k(t+1) = \sum_i \psi_i^k x_i \quad (2.32)$$

### 2.4.3 NG による学習例

NG は、各ユニットが自由に移動することが可能なため SOM のように可視化することは難しい。そこで、入力の次元を 2 として、参照ベクトル（出力）も 2 次元としてマップを作成することで確認を行う。また、SOM と NG を比較することによって NG との性能評価を行う。

学習するデータは、3 つの学習データを用意した。まず、はじめに Fig.2-11 に示す一様分布のデータを使用した。赤色の点がランダムに生成された学習データである。この時、学習した後の SOM 及び NG の参照ベクトル（出力）は、どのように配置されているかを確認する。Fig. 2-12, Fig.2-13 に SOM および NG の結果を示す。赤色で示す点が入力ベクトルであり、青色で示した座標を参照ベクトルのある点とする。これらの結果をみると SOM は、近傍関係が重要なため内挿するように近傍のユニット同士が順序よく整列している。一方で NG の方は、近傍との関係がないため自由に配置されている。また、SOM に比べて広範囲を網羅していることがわかる。同様に、円をえがく一様分布に対し(Fig.2-14)、SOM 及び NG の学習結果を Fig.2-15, Fig.2-16 に示す。ドーナツ型の円に関して、SOM では、ドーナツ上の内側のデータが存在しない部分も補間することができる。しかし、学習データには、円の中心部分のデータはないので、無駄な参照ベクトルがあるということになる。一方で NG では、ドーナツ上に分布しているデータ群を上手く写像していることがわかる。最後に、Fig.2-17 で示すような 2 つの集まりのある一様分布を考えるとする。この時、SOM の結果 (Fig.2-18) では、連続して参照ベクトルが写像しているようになっており、一方、NG の結果 (Fig.2-19) では、独立的に 2 つの一様分布を写像していることがわかる。このことからわかるように両方のアルゴリズムに写像能力があることは証明したが、連続しているものを写像したい場合は、SOM を用いたアルゴリズムが有用であると考えられ、独立的なものを写像したい場合は、NG の方が優れていると考えられる。

## 2.5 モジューラ型自己組織化マップ (mnSOM)

### 2.5.1 モジューラ型自己組織化マップの概要と特徴

モジューラ型自己組織化マップ(modular network SOM: mnSOM)は3節で紹介した T.Kohonen によって考案された SOM の各ユニットを学習可能なネットワーク (ニューラルネットワークなど), つまり機能モジュール(function module)に置き換えたものである[2-19].SOM のアルゴリズムでは, 入力データベクトルしか扱うことが出来ない.しかし, モジューラ型 SOM (mnSOM) では, 内挿補間や位相保持といった SOM の特徴をもったまま, 関数空間なども表現できるといった SOM の拡張したものとなっている.

mnSOM の最大の長所は, 機能モジュールを学習するデータや目的に応じて代えるが出来ることである.また, mnSOM のアルゴリズムはモジュールに依存しないため目的に応じて便宜的に変更できる.

今回は機能モジュールに多層パーセプトロン(Multi layer perceptron: MLP)とした MLP-mnSOM を使用した. MLP は2章で説明したように, 入出力の関係をネットワークで表現することが出来るため, MLP-mnSOM は学習によってできた特徴マップによって関数間の関係を見出すことが出来るだけでなく, 学習完了したネットワークに入力を与えることで何らかの出力をもつシステムをして使える.

### 2.5.2 モジューラ型自己組織化マップの構造

mnSOM の構造を Fig. 2-20 に示す.基本的な構造は, 従来の SOM の各々の格子状に存在するユニットベクトルと同様であり, ユニットベクトルを機能モジュールに置き換えたものである. 機能モジュールは, 格子状にならべてあり特徴マップの座標を表現して並んでいる. Fig. 2-20 には MLP モジュールを機能モジュールには描いている. 各々のモジュールは, モジューラ構造によって '関数の特徴' を表現し, mnSOM 全体によって, SOM と同様に特徴マップを表している.

mnSOM の重要な概念として, われわれははじめに mnSOM の使用者がある初期状態のわからない関数グループ  $\{f_1(x_1), \dots, f_l(x_l)\}$  のマップを作ることを一つ

の例として始める.その際に, サンプルデータ  $\{D_1, \dots, D_I\}$  は与えられていると仮定する.ここに,  $D_i = \{(x_{i,1}, y_{i,1}), \dots, (x_{i,j}, y_{i,j})\}$  は,  $i$  番目からサンプリングされた入出力ベクトルのクラスである.その上, このベクトルのペアは,  $y_{i,j} = f_i(x_{i,j})$  を満たす. mnSOM の仕事は, (1)未知の関数を同定することと(2)それらの関数の特徴マップを生成することである.つまり, mnSOM の仕事は, 関数空間からマップ空間までのトポロジカルなマップを写像しているのである(Fig.2-21). それゆえに mnSOM はベクトル空間よりもむしろ関数空間の SOM であるべきであるといえる.

### 2.5.3 mnSOM の学習アルゴリズム

今回のアルゴリズムの説明は, MLP モジュールの場合の MLP-mnSOM(MLP-mnSOM)について考える. MLP-mnSOM は  $K$  ファンクション  $g^1(x_{i,j}), \dots, g^K(x_{i,j})$  によって表現される  $K$ -MLP モジュールを持っていると仮定する.これらの関数は, バックプロパゲーションのアルゴリズムによって更新された重みベクトル  $w^1, \dots, w^K$  によって決定される.本論文では, マップオブジェクトおよびデータセットのインデックスは下付き文字で表し, mnSOM のモジュールのインデックスは上付き文字で表す.

mnSOM のアルゴリズムは4つのプロセスによって成り立っている.それは, 評価プロセス, 競合プロセス, 協調プロセスと適合プロセスである. 以下にその過程を述べる.

#### 2.5.3.1 評価プロセス

評価プロセスではすべての MLP モジュールからの出力は, 各々の入出力ベクトルのペアから評価される.例えば, 入力データベクトルを  $x_{i,j}$  としたとき,  $k$  番目のモジュールの出力  $\tilde{y}_{i,j}^k = g^k(x_{i,j})$  は入力から計算される.同様に入力  $x_{i,j}$  を使用して  $k=1, \dots, K$  までモジュールを使用して繰り返される.すべての入力について出力が評価された後, すべてのモジュールは各々のデータクラスとすべてのモジュールのエラーを評価する.そして,  $E_i^k$  は例えば,  $i$  番目のからのデータセットの  $k$  番目のモジュールのエラーを表している.

$$E_i^k = \frac{1}{J} \sum_{j=1}^J \left\| \tilde{y}_{i,j}^k - y_{i,j} \right\|^2 \quad (2.33)$$

$$= \frac{1}{J} \sum_{j=1}^J \left\| g^k(x_{i,j}) - f_i(x_{i,j}) \right\|^2 \quad (2.34)$$

もし、 $J$  (サンプルの数) が十分に大きいときは、関数空間の  $k$  番目のモジュールと  $i$  番目のシステムとの距離は、以下にあらわすエラー  $E_i^k$  によって概算される。

$$L^2(g^k, f_i) = \int \left\| g^k(x) - f_i(x) \right\|^2 p_i(x) dx = E_i^k \quad (2.35)$$

ここで、 $p_i(x)$  は  $i$  番目のシステムの入力データベクトル  $x_{i,j}$  の確率密度関数である。本論文では、 $i = 1, \dots, I$  である  $\{p_i(x)\}$  はおおよそ  $p(x)$  と各々のクラスによってデータの表現を正規化しているため同様であると仮定できる。

### 2.5.3.2 競合プロセス

各データ集合  $D$  に対して、エラー  $E_i^k$  の最小であるモジュールをベストマッチングモジュール(BMM)として決定する。例えば、 $i$  番目のモジュールを勝者とする、 $i$  番目のクラスのベストマッチングモジュールのモジュール番号を  $k_i^*$  とする。このとき  $k_i^*$  は、

$$k_i^* = \arg_k \min E_i^k \quad (2.36)$$

と表される。

### 2.5.3.3 協調プロセス

協調プロセスは、学習率  $\psi_i^k$  を近傍関数  $h(l;t)$  を用いることで計算する。

$$\psi_i^k(t) = \frac{h(\|\xi^k - \xi^{k_i^*}\|; t)}{\sum_{i=1}^I h(\|\xi^k - \xi^{k_i^*}\|; t)} \quad (2.37)$$

$$h(l;t) = \exp\left[-\frac{l^2}{2\sigma^2(t)}\right] \quad (2.38)$$

ここで、 $\xi^k$  はマップ空間での  $k$  番目のモジュールの位置を示しており、そして  $h(l;t)$  は、時間  $t$  のときに計算して縮小させた近傍半径のことである。

### 2.5.3.4 適応プロセス

適応プロセスでは、MLP モジュールのすべては以下のバックプロパゲーション学習アルゴリズムによって学習させたものである

$$\Delta w^k = -\eta \sum_{i=1}^I \psi_i^k(t) \frac{\partial E_i^k}{\partial w^k} = -\eta \frac{\partial E^k}{\partial w^k} \quad (2.39)$$

(2.39)式はすべてのモジュールが十分に更新されるまで繰り返される。例えば、バックプロパゲーション学習は繰り返し時間  $t$  ごとに数千回繰り返すことも可能である。ここで、 $E^k$  は、 $E^k = \sum_i \psi_i^k E_i^k$  によって決定される  $k$  番目のモジュールのエラーを表している。ここで注目すべき点は、 $E^k$  は  $g^k(x)$  による全体的な極小点を持っていることである。つまり、 $g^k(x)$  は

$$g^k(x) = \sum_{i=1}^I \psi_i^k(t) f_i(x) \quad (2.40)$$

によって計算される。それゆえに  $g^k(x)$  は重み  $\{\psi_i^k\}$  による  $\{f_i(x)\}$  の内部をバックプロパゲーションによって学習させている。MLP の訓練の間、各々の入力ベクトル  $x_{i,j}$  は、入力を一つずつ与えられ、該当する出力  $y_{i,j}$  は教師信号と比較され更新される(第2章参照)。これらの4つのプロセスを繰り返すことによって、ネットワークが安定したところへ落ち着くまで近傍半径を縮めていく。

## 2.5.4 ローカルミニマムの回避方法

MLP-mnSOM にとって、ローカルミニマムは、各モジュールがローカルミニマムに入ってしまうとマップの各々の点の連続性が破壊されてしまうため深刻な問題である。さらに悪いことに、モジュールの数が増えるということは、ローカルミニマムに落ちるチャンスを増やすことになるということである。それゆえにローカルミニマムを回避する追加のアルゴリズムが必要となる。

(2.39)式に示したように、 $k$  番目のモジュールの仕事はエネルギー関数である  $E^k = \sum_i \psi_i^k E_i^k$  を最小限にすることである。これは、 $M^k$  においてすべてのモジュールの間で  $E^k$  が最小となることを期待されていることを意味している。このモジュール  $M^k$  が他のモジュール  $E^k$  に一時的に変わると仮定する。もし、エネルギー関数  $E^k$  がこの置換によって改良されるのであれば、それは  $M^k$  がローカルミニマムの問題を回避することが出来る。つまり、 $M^k$  は関数モジュール  $g^{k'}$  に  $g^k$  をコピーすることで‘救助’することが出来る。このアルゴリズムを明確に示すと以下の式となる。

$$g^k(t+1) = g^{k'}(t) \quad \text{if} \quad \tilde{E}^{k,k'} < \beta E^k \quad (2.41)$$

$$\tilde{E}^{k,k'} \triangleq \sum_{i=1}^I \psi_i^k E_i^{k'} \quad (2.42)$$

ここで  $\tilde{E}^{k,k'}$  は、 $M^k$  が一時的に  $M^{k'}$  と代えられたときのエネルギー関数であり、 $\beta$  は普段 0.8~0.9 付近を普段使う安全項である。この追加アルゴリズムを採用することによって、ローカルミニマムの中の落ちる可能性はモジュールが増えることによって助けられる可能性が増えるため減少するであろう。それゆえにこのアルゴリズムは多くの MLP の持つ欠点を利点へと変化させるものである。このアルゴリズムはすべてのモジュールの構造においてローカルミニマムの問題に対して効果的である。

### 2.5.5 MLP-mnSOM による学習例

MLP-mnSOM の学習例として、三次関数  $y = ax^3 + bx^2 + cx$  について行った。クラスごとに複数のデータ点 ( $J=21$ ) をもったクラス ( $I = 6$ ) をもちいた。各々のデータ列 (クラス)  $D_i = \{(x_{i,j}, y_{i,j})\}$  は 0.2 間隔で  $[-1,+1]$  の範囲で値を得た。各種パラメータに関しては Table 2-4 に示す。実際に与えた学習クラスは、0, 1 番目のクラスは 1 次関数, 2, 3 番目のクラスは 2 次関数, 4, 5 番目のクラスは 3 次関数を表現したデータ点の集合である。実際に mnSOM の結果を以下の Fig. 2-22 に示す。

Fig. 2-22 をみて分かるように学習初期は、一次関数を表現したモジュールが多いが学習が次第に進むにつれて 6 つのクラスの関数を表現していることが分かる。Fig.2-23 に示すように各次元の関数はマップ上左右対称の部分におおよそ分布され、上に凸 ( $a$  の係数が-) と下に凸の部分 ( $a$  の係数が+) に分類される。また、学習する際に BMM となったモジュールを赤の枠で囲っている。このように BMM の関数を写像することによって、他のモジュールもそれに近似するような関数を表現していることがわかる。以上のように mnSOM では、SOM では表現できなかった関数空間の写像が可能となる。

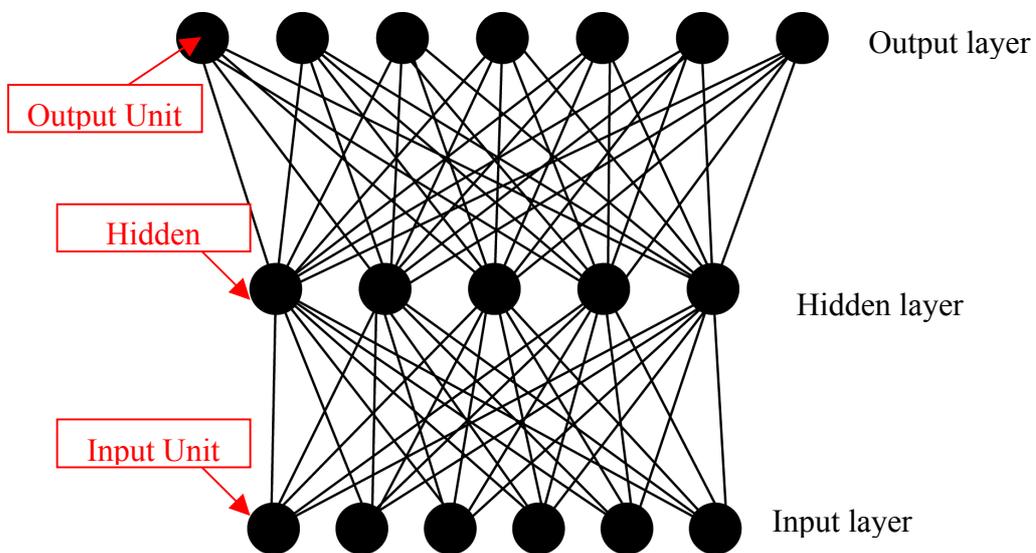


Fig.2-1 Model of Multi Layer Perceptron

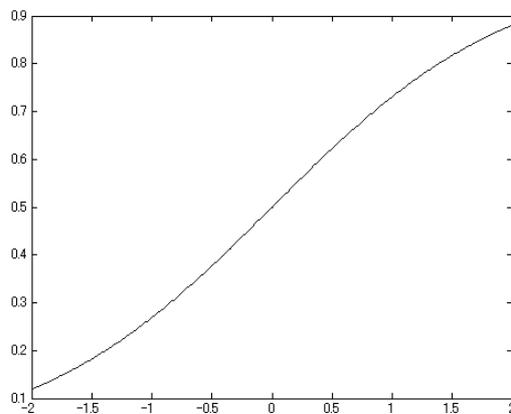


Fig.2-2 Graph of sigmoid function

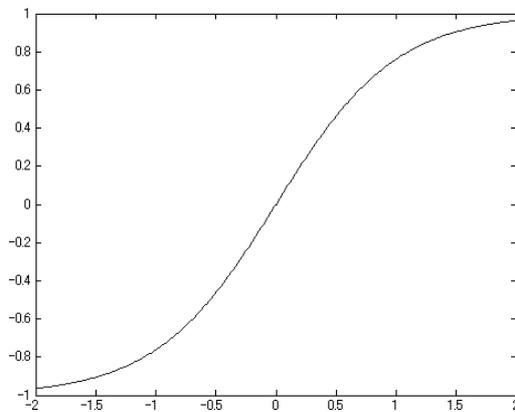


Fig.2-3 Graph of hyperbolic

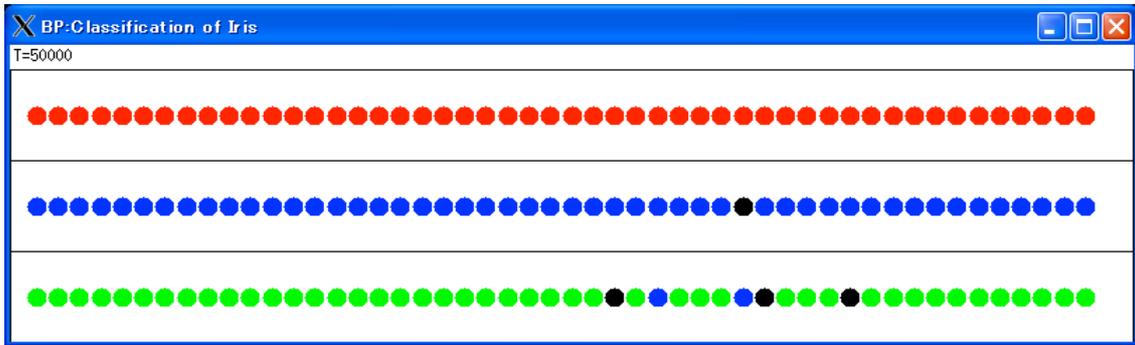


Fig.2-4Result of BP learning

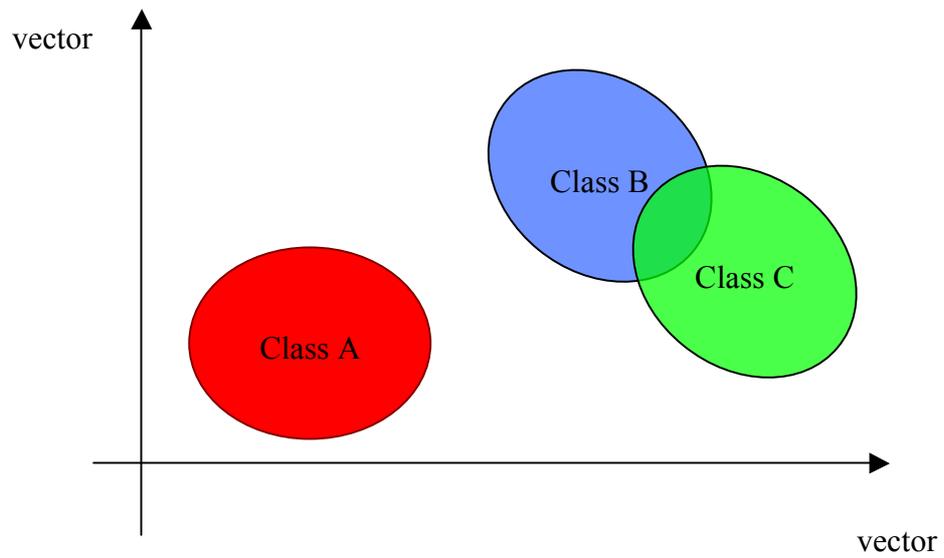


Fig.2-5 Vector space of Iris

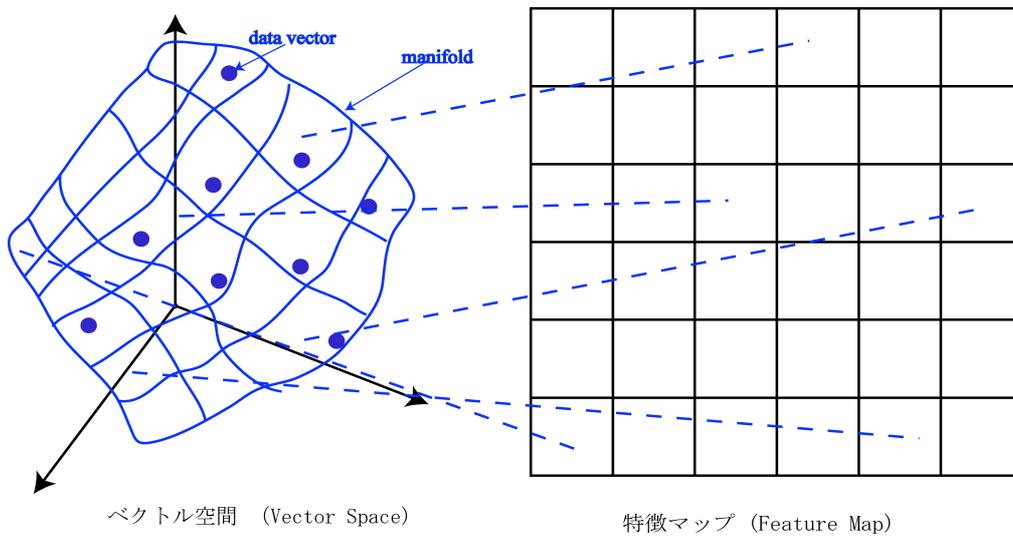


Fig. 2-6 Vector space and feature map

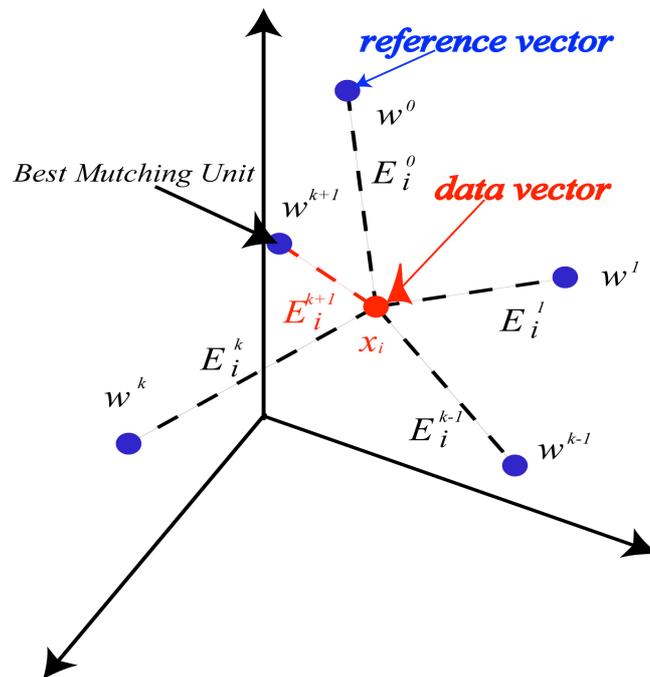


Fig.2-7 Error between data vector and reference vectors

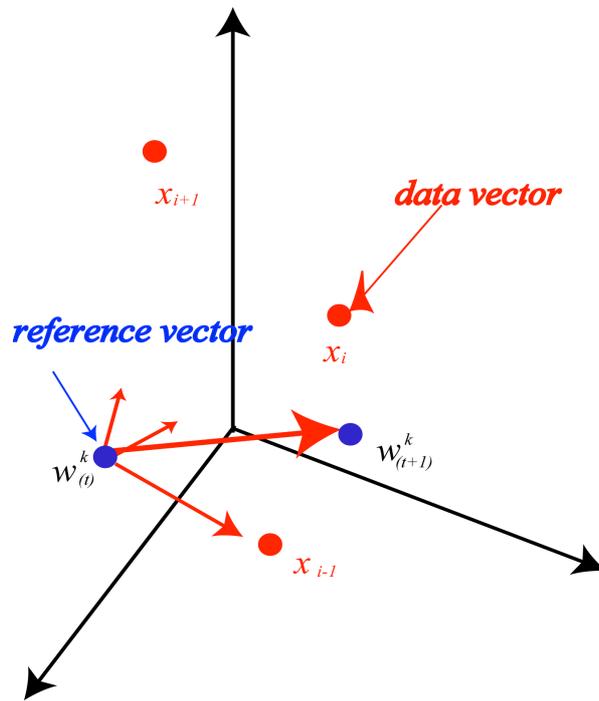
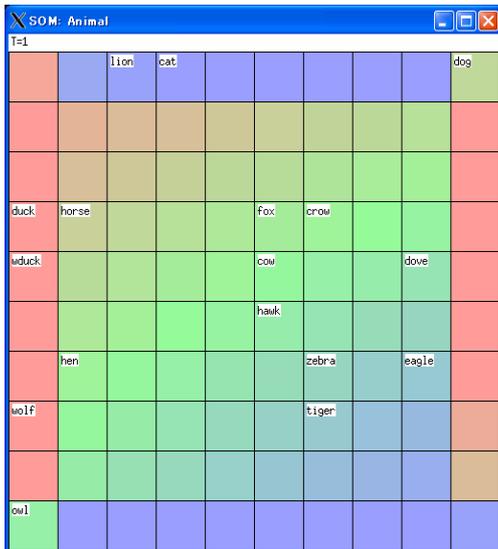
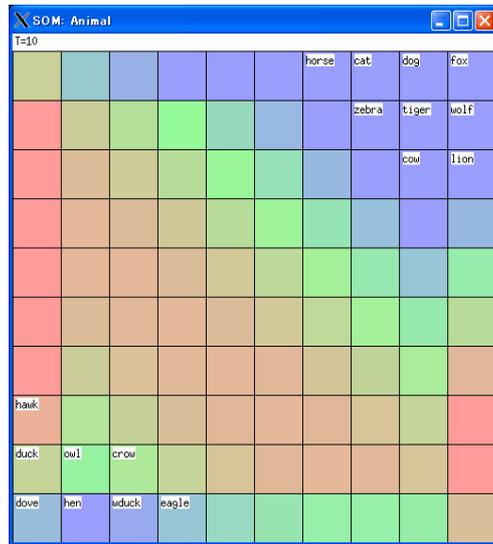


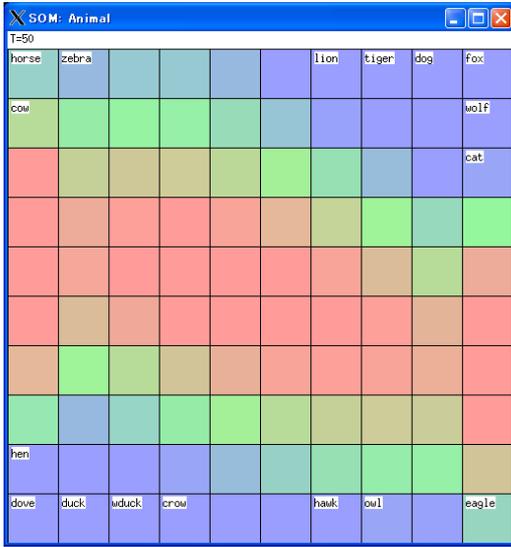
Fig. 2-8 Adaptation of reference vectors



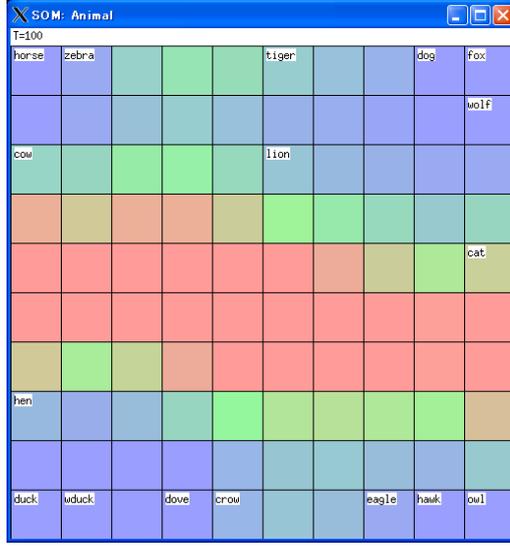
(a) t = 1



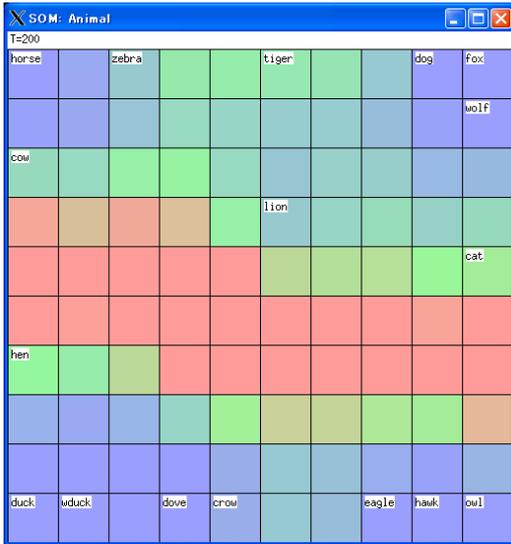
(b) t = 10



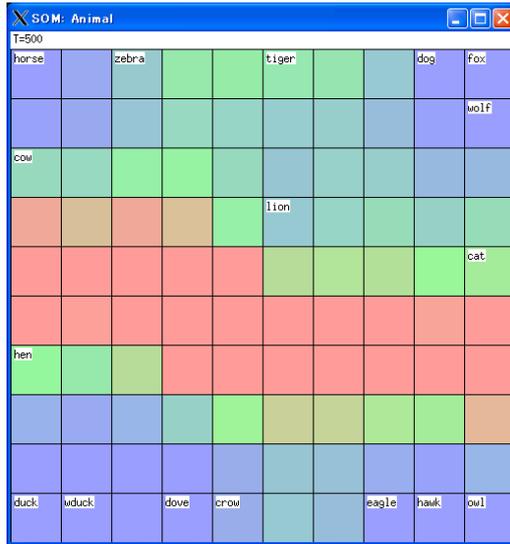
(c)  $t = 50$



(d)  $t = 100$



(e)  $t = 200$



(e)  $t = 500$

Fig. 2-9 change of feature map by batch learning

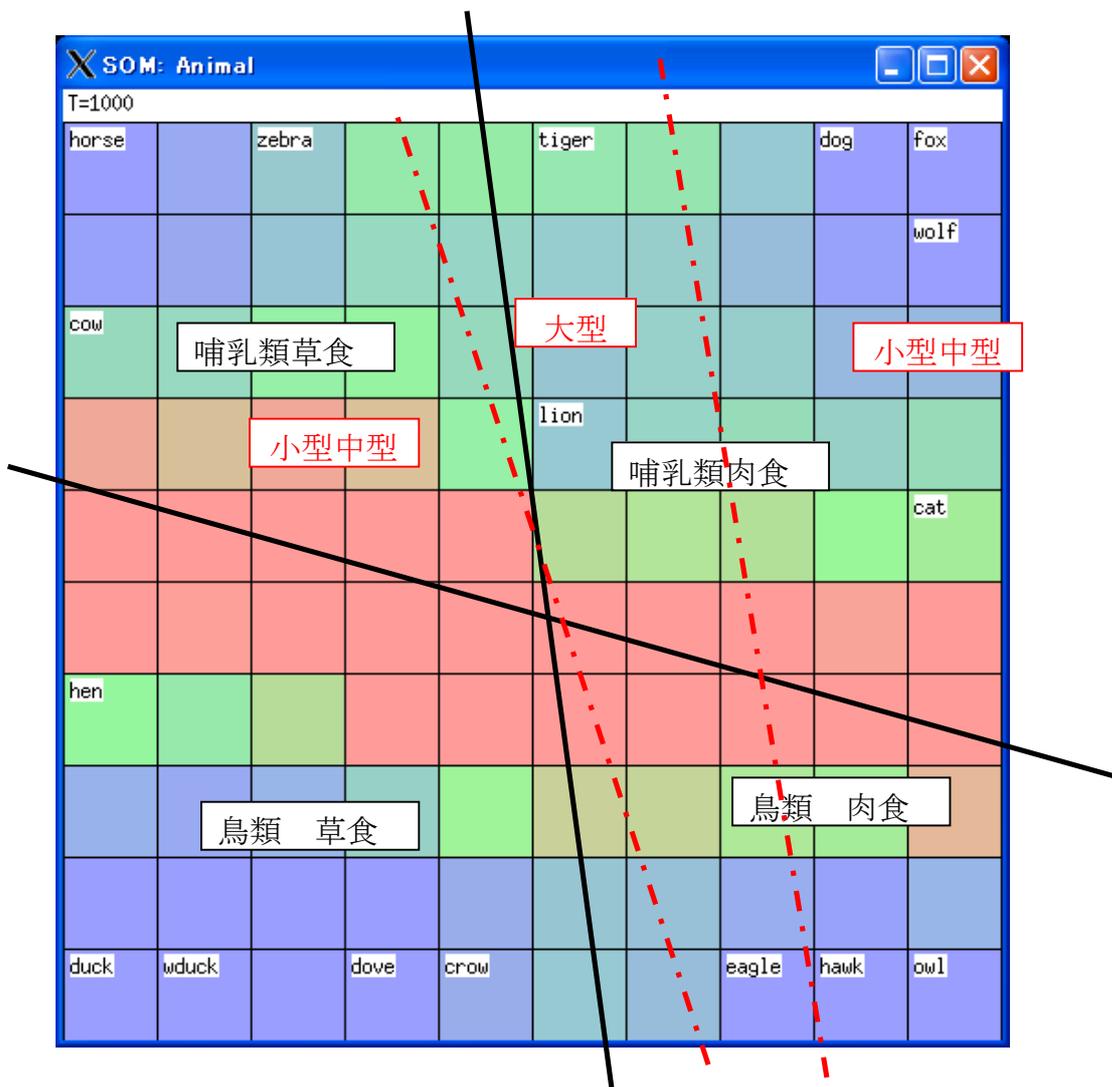


Fig.2-10 The result of Batch SOM feature map

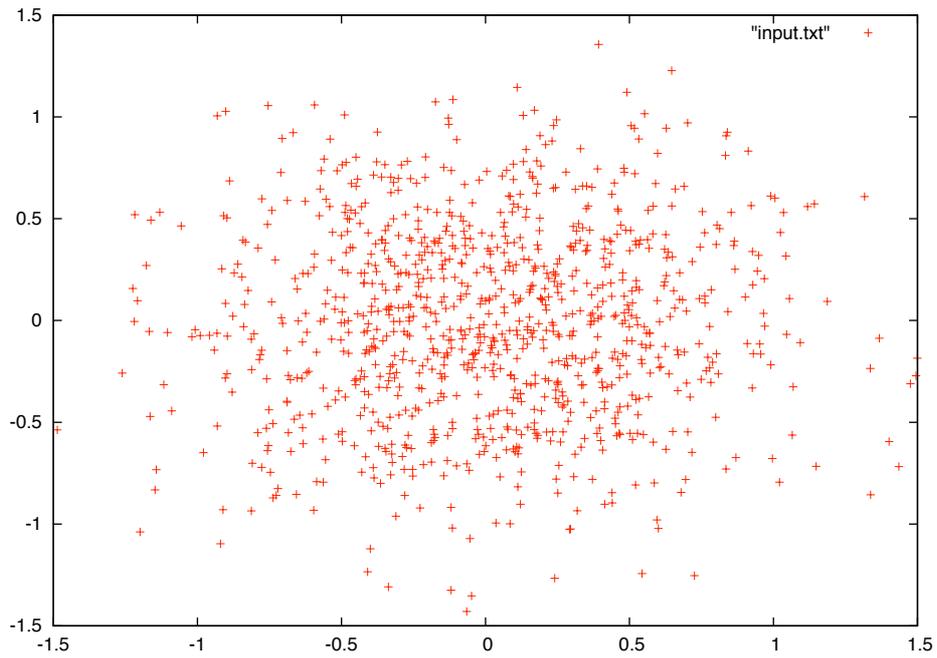


Fig.2-11 an uniform distribution learning data

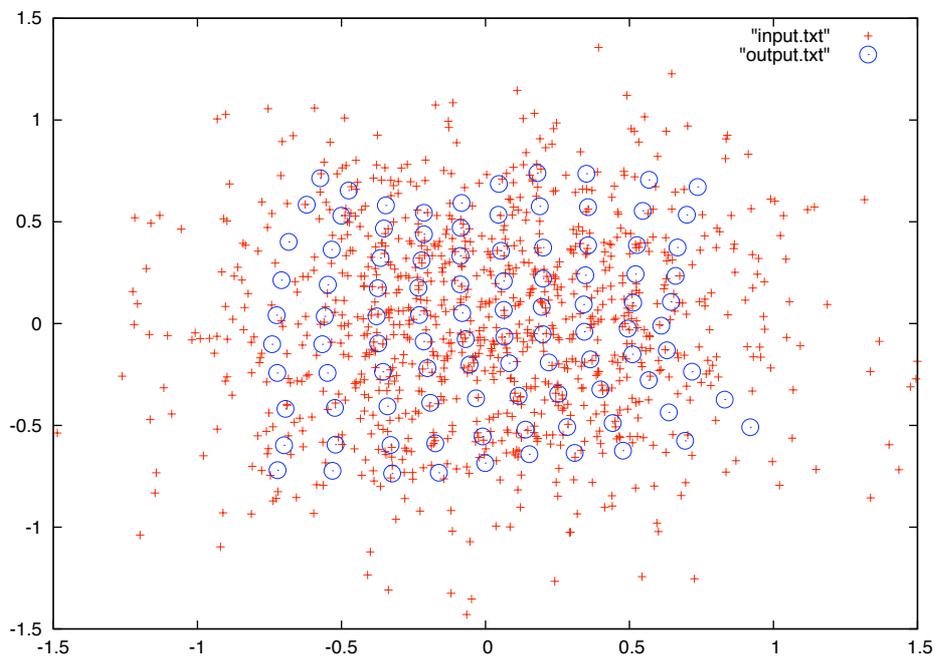


Fig.2-12 The result of the SOM algorithm (an uniform distribution)

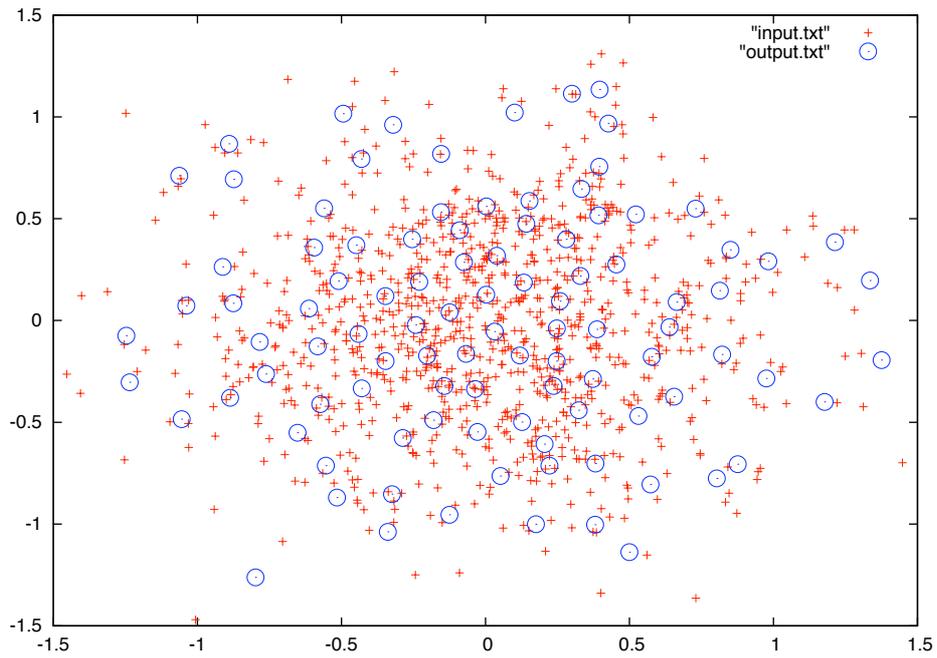


Fig.2-13 The result of the NG algorithm (an uniform distribution)

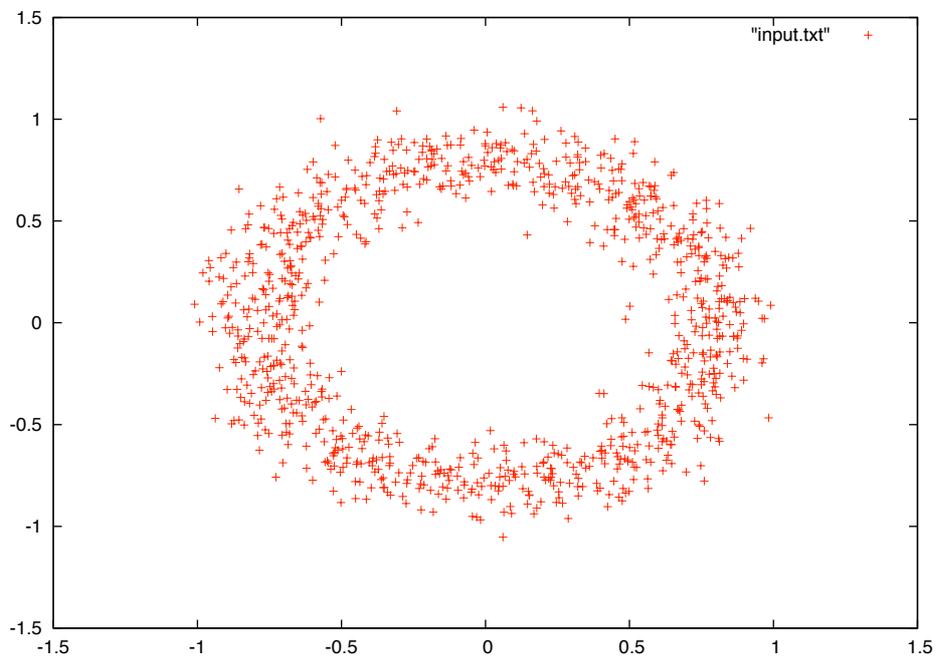


Fig.2-14 the Circle (a doughnut) learning data

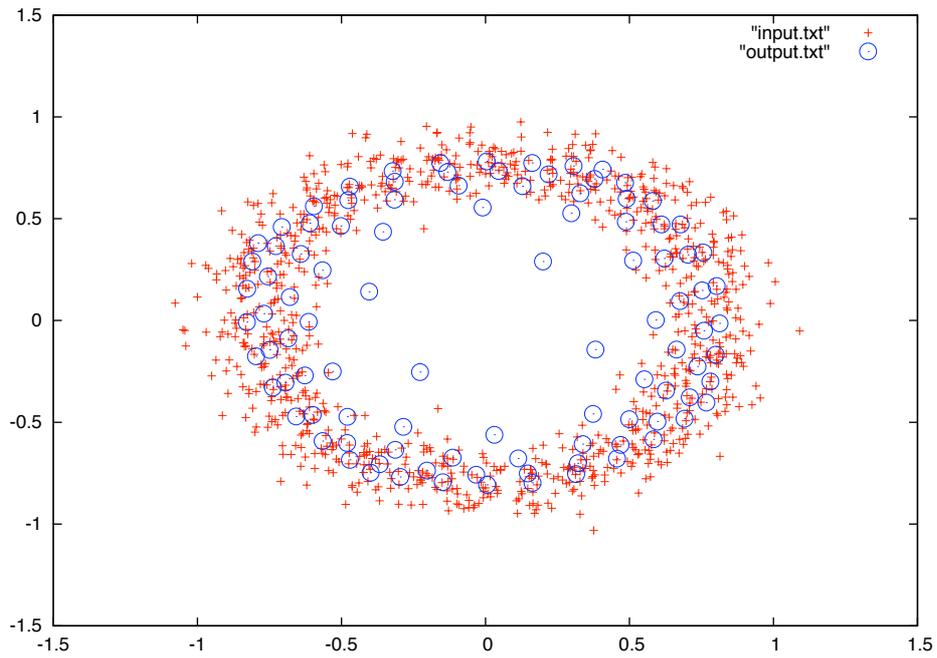


Fig.2-15 The result of SOM algorithm (a doughnut)

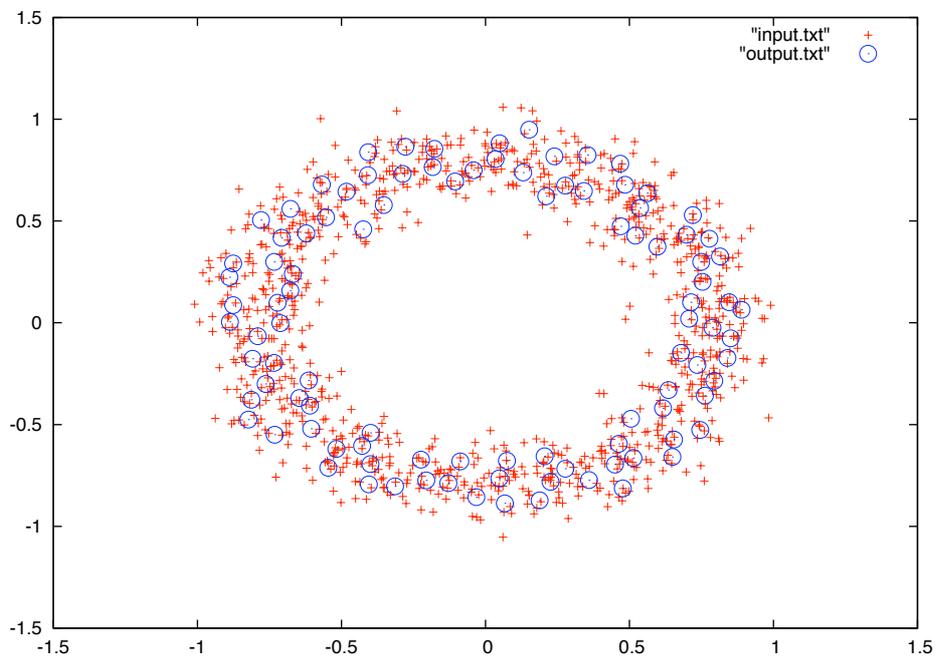


Fig.2-16 The result of NG algorithm (a doughnut)

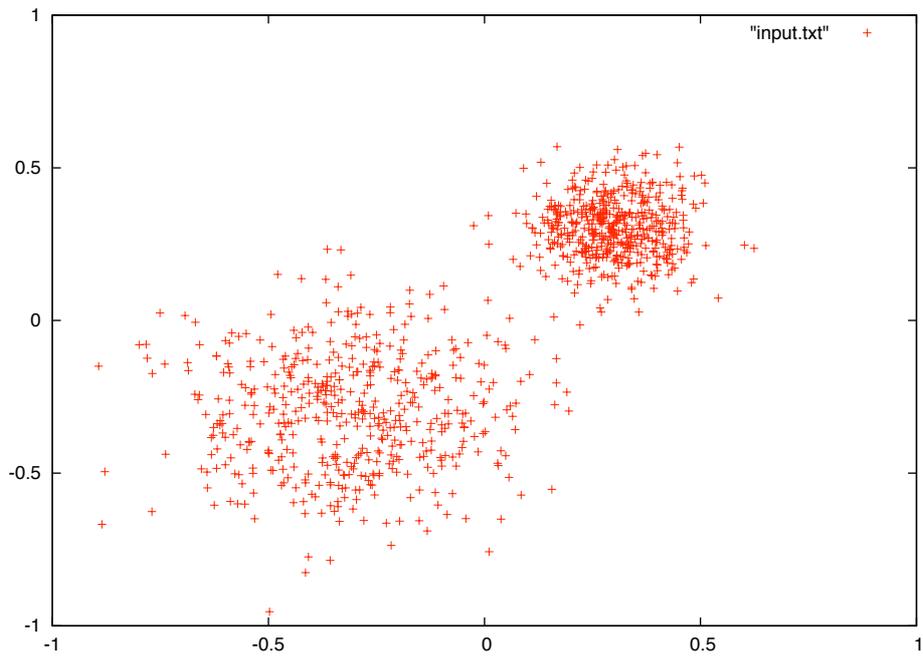


Fig.2-17 The two uniform distribution learning data

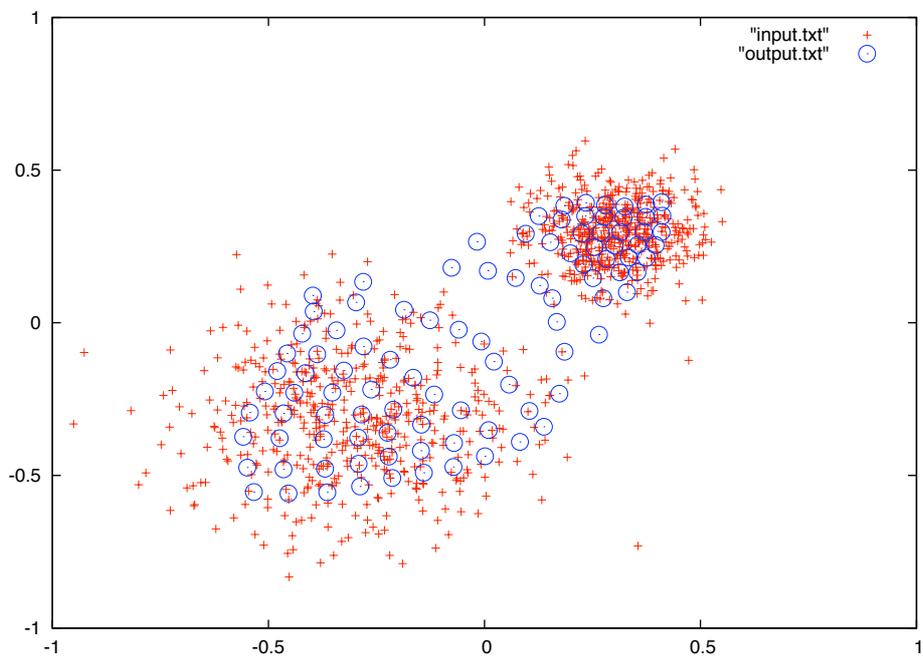


Fig.2-18 The result of SOM algorithm (two uniform distribution)

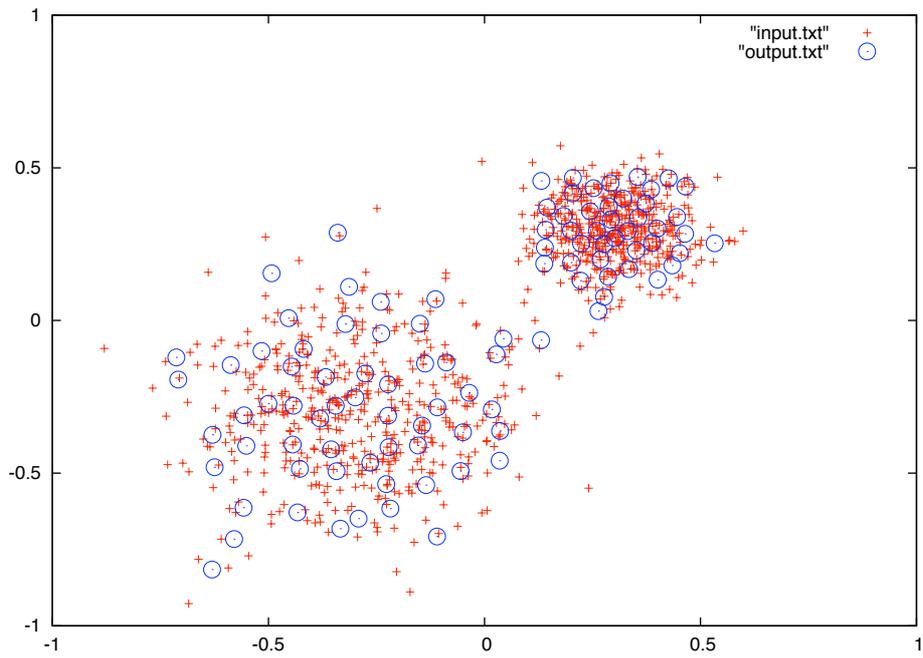


Fig.2-19 The result of NG algorithm (two uniform distribution)

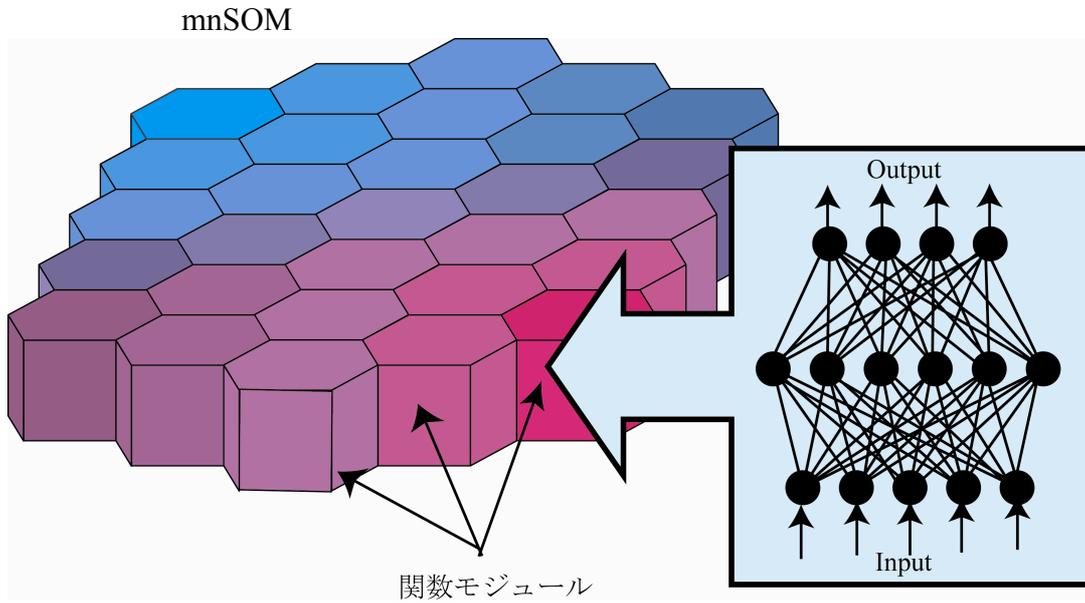
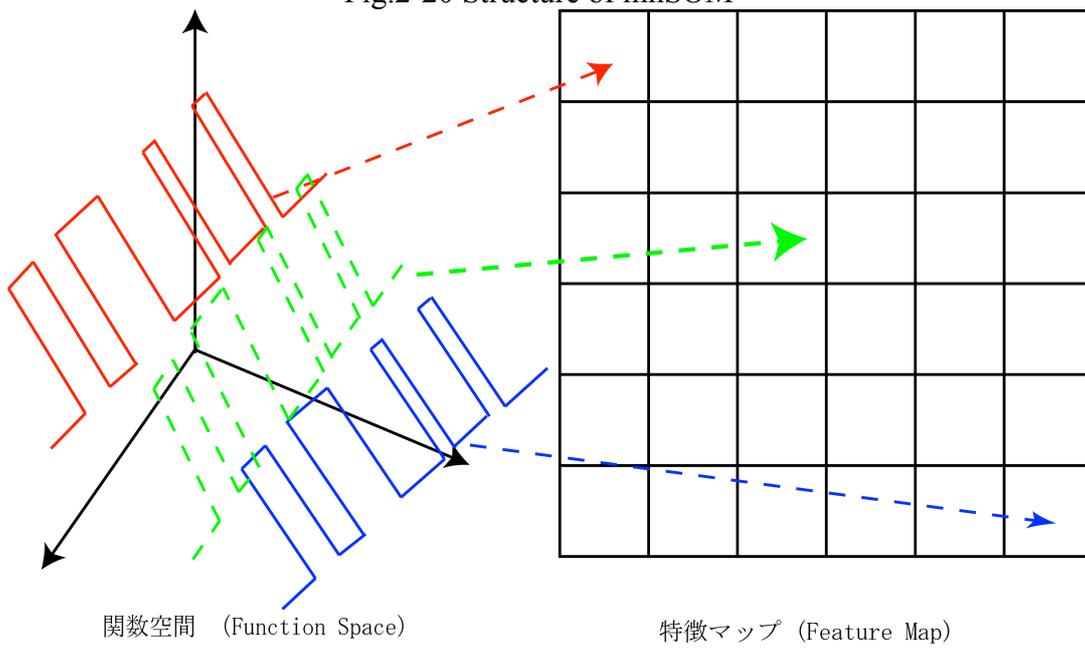
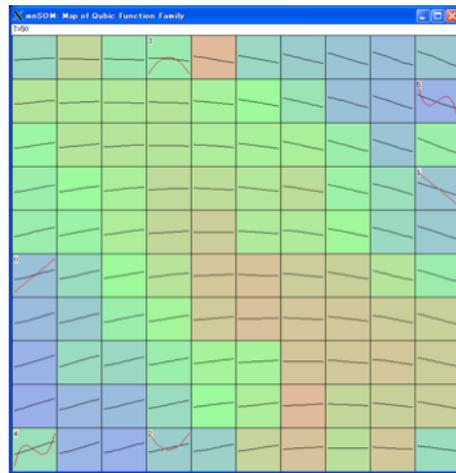


Fig.2-20 Structure of mnSOM





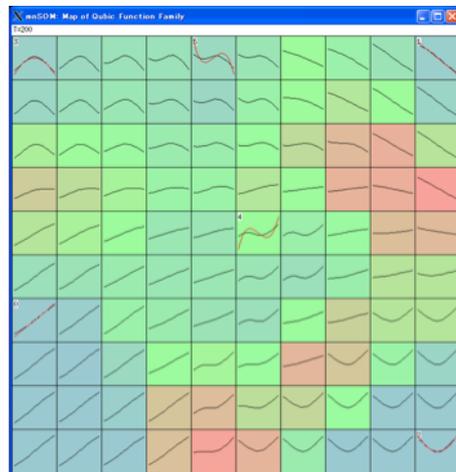
(a)  $t = 0$



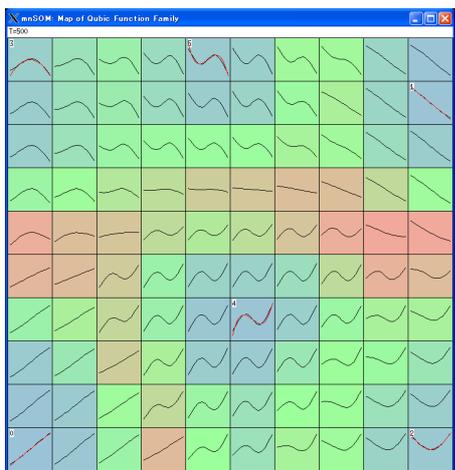
(c)  $t = 50$



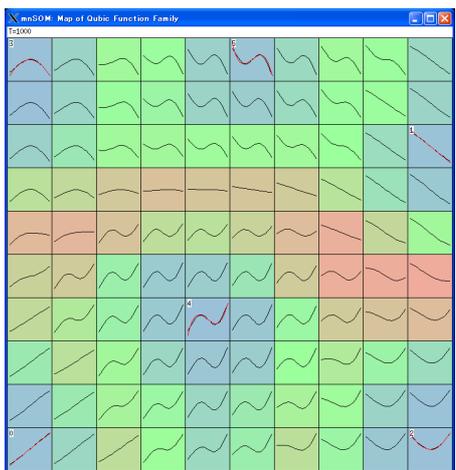
(c)  $t = 100$



(d)  $t = 200$



(e)  $t = 500$



(f)  $t = 1000$

Fig.2-22 Result of cubic function mnSOM

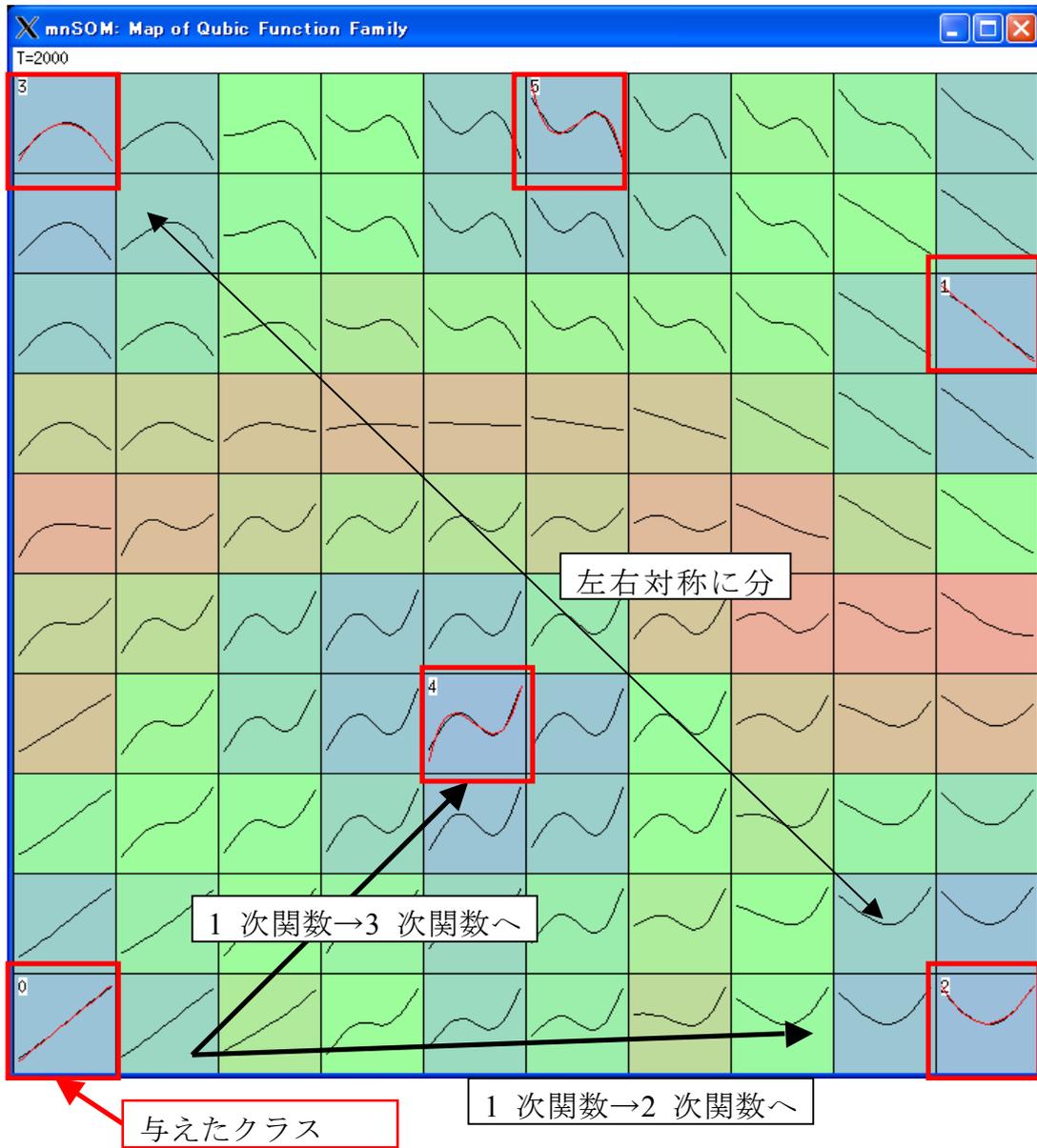


Fig.2-23 Result of mnSOM feature map

Table 2-1 Variable used explanation of SOM algorithm

$x$	学習データ
$i$	クラスを表す添え字 ( $i = 1, \dots, D$ )
$w$	参照ベクトル
$k$	ユニット番号を示す添え字 ( $j = 1, \dots, J$ )
$E$	学習データと参照ベクトルの距離
$k^*$	ベストマッチングユニット
$\phi$	近傍関数
$d(a,b)$	$a, b$ 間のユークリッド距離
$\psi$	学習率
$\sigma$	近傍半径
$\tau$	時定数

Table 2-2 Animal data vectors

	ハト	メンドリ	アヒル	カモ	カラス	フクロウ	タカ	ワシ	キツネ	イヌ	オオカミ	ネコ	トラ	ライオン	ウマ	シマウマ	ウシ
小型	1	1	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0
中型	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
大型	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
夜行性	0	0	0	0	0	1	0	0	0.5	0	1	0.5	0.5	0	0	0	0
2本足	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
4本足	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
毛をもつ	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
蹄をもつ	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
髭をもつ	0	1	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0
羽をもつ	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
縞	0	0	0	0.3	0	0	0	0	0	0	0	0	1	0	0	1	0
狩猟	0	0	0	0	0	1	1	1	1	0	1	1	1	1	0	0	0
早く走る	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	0
飛ぶ	1	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
泳ぐ	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
家畜	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1
草食	1	1	1	1	0.5	0	0	0	0	0.5	0	0.5	0	0	1	1	1
肉食	0	0	0	0	0.5	1	1	1	1	0.5	1	0.5	1	1	0	0	0
イヌ科	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
ネコ科	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0
ペット	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0

Table 2-3 Parameters of SOM algorithm

ユニット数	10x10
最小近傍半径 $\sigma_{\min}$	2.0
最大近傍半径 $\sigma_{\max}$	15.0
時定数 $\tau$	50.0

Table 2-4 Variable used explanation of NG algorithm

$x$	学習データ
$i$	クラスを表す添え字 ( $i = 1, \dots, I$ )
$w$	参照ベクトル
$k$	ユニット番号を示す添え字 ( $j = 1, \dots, J$ )
$E$	学習データと参照ベクトルの距離
$k^*$	ベストマッチングユニット
$\phi$	近傍関数
$d(a,b)$	$a, b$ 間のユークリッド距離
$\psi$	学習率
$\sigma$	近傍半径
$\tau$	時定数

# 第3章

## 知能化の基礎

### (ハードウェアの知能化)

### 3.1 ハードウェアの知能化

第2章では、ソフトウェア的なアプローチからロボット知能化を行う為に脳型情報処理アルゴリズムを用いる手法について述べた。第3章では、ハードウェア的なアプローチからのロボットの知能化を考える。ロボットが行動する際に、高度な知能を持って行動する為には、プラットフォーム自体の性能も大きく関わってくる。つまり、ハードウェアの設計上の段階に置いて、操作性、拡張性、信頼性、安全性等を考慮して設計しなければ、ソフトウェアへの開発比重が大きくなる。つまり、ハードウェアの性能は、ソフトの複雑な設計とのトレードオフな関係が成り立つ。そこで、我々は、ハードウェアについても知能化する必要があると考えた。ハードウェアの知能化の定義を1章で示すように以下に示す。ハードウェアの知能化は、**ロボットを設計する段階で性能を最大限に活かせるように設計する指針の事**と定義する。

そこで、本研究では、アミールによって提唱された“Concept of Intelligent Mechanical Design for Autonomous Mobile Robots”[1-3]を参考にロボットのハードウェアの知能化を行う。本章では、アミールによって提唱されたロボットの知能化されたデザイン手法について述べる。

### 3.2 Intelligent Mechanical Design

自律型移動ロボットを設計・開発を行う上で、デザインには様々な要求が存在する(Fig.3-1)。例えば、Fig.3-1に示すように単純な機構であることや信頼性、拡張性、安全性といった様々な要求が設計段階で考えなければならない。そして、ロボットを動かす環境やロボットがどのような事を行うかなどがロボットを設計する為に必要な条件として挙げられる。そこで、本研究では、これらの様々な要求に対して、ハードウェアの知能化(メカテリジェント: Mecha-telligent)を行う事で、要求に対するデザイン手法を確立する(Fig.3-2)。

アミールによって示された設計指針によると、Fig.3-3に示す設計指針は、ロボットに要求されるいくつかの項目を明らかにする事でロボットを設計する事が出来る。以下にアミールによって示された設計指針に従って、自律型移動ロボットを開発する上で必要な項目を考える。

まず、基本項目として、設計するロボットの基本的な要求を考える。このとき

ロボットに基本的に必要な設計上の項目を以下に挙げる.

- (1) ロボットのタスク
- (2) 必要な行動
- (3) 使用する環境
- (4) 環境による拘束条件

以上の条件より, 様々なロボット形態がある中で開発するプラットフォームにあった移動形態を採用する. そして, ロボットのハードウェアの知能化を果たす為に次節で示す方針を基本項目を参考に組み立てる.

### 3.3 Mechanical Design Principles: “Mecha-telligence Principles ”

“ロボットのハードウェア上の知能化” は, どうやって機械的な構造を効果的に, 操作性や単純性, 拡張性を考慮して, デザインする事が出来るかということを示したものである. このデザイン手法は, 7つのデザインの原則を考慮する事によって構成されている. このデザイン手法をメカテリジェンス手法と呼ぶ. Fig.3-3 で示すようにメカテリジェンス手法は, 自律的に制御可能で, 信頼性のある, 現実可能な互換性を持つ設計に関して問題を考える時に問題を解決する為の手順を示している. 以下に7つのデザイン原則の詳細を示す.

#### Principle 1: The three-constituents principle

自律型移動ロボットの機械部分を設計するには, 3つの要求が関係する.

- (1) 環境による要求
- (2) 必要なロボットの行動とタスク
- (3) ロボット運動学的知見からの条件

これを主要な要求としてまず考えなければならない. つまり, 環境による要求に対して適した機械部分を設計しなければならない. そして, ロボットの行動とタスクを十分に理解して, それに必要なロボットの構成を考えなければならない.

## Principle 2: The complete-mechanical design principle

完全に要求を満たす機械のデザインの原則は、ハードウェア的な知能化による設計手法が完全に行われたときであると考えられる。そこで、まず、信頼性や実現可能性、互換性、拡張性等が十分に満たされているかを考える。

## Principle 3: The principle of mechanical balance

機械設計に置いて、バランスを考えるときには2つの事を考えなければならない。最初に、機械のデザインの複雑さがロボットの望ましい要求を満たす上で必要な複雑さであるのかどうかである。つまり、ロボットに必要以上の要求を満たす事は、バランスが崩れてしまう。よって、ロボットに必要最小限のものを考え、バランスを考えながら設計を行わなければならないという事である。次に、ロボットの機構部分を不明瞭のままデザインをしないようにしなければならないということとである。つまり、センサ等の仕様を考えるときに環境等の要求に対して必要なものを取り入れ設計を行っていくという事である。

## Principle 4: The environmental principle

環境についての原則は、基本的に2つある。一つ目は、システムと環境自体の相互作用を考える事を意味する。つまりシステムが環境自体にあった運動学的性質を利用し設計されなければならない事を示している。次に、環境による制約条件を利用する事である。例えば、ロボットの競技会等では、ルールなどが明確に設定されており、ロボットに対しての拘束条件として大きく設計に関わってくる。そのため環境による拘束条件を考慮して、設計する事が重要である。

## Principle 5: The sensory-actuator principle

センサとアクチュエータに対する原則としては、センサの位置と種類がロボットのアクチュエータを動かす機構に対して、直接的に効果的である事である。その為に、センサの種類や数は、単純で必要最小限なものであり、信頼性のあるものでなければならない。その為、センサの選定の段階でロボットのどこに設置する事でロボットのパフォーマンスを向上させる事が出来るかも考えなければならない。つまり、センサとアクチュエータのつながりを設計段階において考えておく事で、ロボットは直接的にセンサの機能を効果的に取り入れる事が可能となると考えられる。

## Principle 6: The principle of sharing sensory-actuator system

センサとアクチュエータの共有原則では、ロボットに必要以上のセンサを積み込まないように考える事である。つまり、Principle 5 によって示した原則より直接的に必要なでないセンサに関しては極力のぞく事を意味している。これによってロボットの単純性を保つ事が出来る。しかし、インターフェースや電源部分等では、拡張性を持たせてセンサが搭載しやすい設計にする事が重要である。

## Principle 7: The Mechatronics modularity principled

モジュラリティー原則とは、ロボットの電氣的な組み合わせなどを考慮してシステム全体をモジュール構造で設計するという手法である。この原則は Principle 2 における完全な設計手法における信頼性やメンテナンス性の向上を考える上で重要な部分である。以下にその際に行う 3つの段階を示す。

*Step 1: Description of the Robot System Architecture*

*Step 2: Defining the possible module (single module)*

*Step 3: Merging the single module*

最初にロボットの構造を考え、ロボットの機構として似ているものやインターフェースの近いものを明確にする。次に、それらをグループ分けしてシングルモジュールとして設計する。最後にそのシングルモジュールを組み合わせる事によってロボット完成する事が出来る構造となっている。

### 3.4 Mechanical Design Process: “Mecha-Telligence Methodology”

本節では、メカテリジェンス手法としてどのように自律型移動ロボットの機構を設計するかという新しい手法を提案する(Fig.3-4 を参照). 提案する手法の目的は、要求された環境においてロボットと必要なタスクなどの条件 (Principle 1~7 によって定義)をもとにして、ロボットを設計していく手法である. まず Principle 1 によって示された重要度の高い要求をもとに具体的な仕様の形態に段階的に要求に対する仕様を具体化していく. 実際には、重要度の高い仕様では、自律性、移動性能を考え、それ以外にロボットの必要なタスクから導き出される目的 X を軸に設計は行われる. そして、それに対するサブレイヤー (具体化された仕様の形態) を構築し、ロボットが Principle 1~7 で必要とされたものに対するものを具体化していく. サブレイヤーは階層化されており、環境に応じたロボットの形態を考慮しながら構築されていきます. この階層化を繰り返し、前節で示した、Principle 1~7 を取り入れる事で、具体的にロボットの仕様を決定していく. 最終的に仕様として決定したレイヤーをモノスペックレイヤーとして、具体的なロボットに適応したセンサやアクチュエータを決定する. この階層化による要求の具体化を行う事で様々な用途に対しても、同じ方法論に基づいて目的や環境の違うロボットのハードウェアの設計を達成する事が可能である.

[3-1,2,3,4]

### 3.5 ハードウェアの知能化例：サッカーロボット

本章では、全方位移動型サッカーロボットのハードウェアの知能化を目的とする。開発例として、ロボカップサッカー中型リーグにおける基本課題を研究題材と設定し、本研究の目標を据える。

ロボカッサッカー中型リーグは、完全自律分散型の移動ロボットの協調行動問題の良いテストベッドであり、基本的な課題が多く盛り込まれている。まず、基本技術の中核は、移動能力と視覚が必要である。ロボットが自律的に行動し、サッカーを試合時間の間常に行動させるようにするには、頑強であり高い移動能力を備えたハードウェア構成を備えていなければならない。また、サッカーという競技を行うため、ロボット同士の接触なども多く、故障や不具合がおこることが予想される。その為、ロボットのメンテナンス性や操作性がハードウェアの必要な課題となっている。さらに、ロボカップでは2050年の最終目標に向けて、様々なルール変更が毎年行われており、その課題に対して柔軟に対応できる為の拡張性がロボットには必要である。その為、ハードウェア面では、ロボットがこれまでに挙げた要求を満たす為の構造を開発する事が目標であり、過去に開発されたロボットの欠点を検証し、アミールらによって提案された自律型移動ロボットにおける機械的知能化デザイン手法に基づいた[3-2,3]ロボットの知能化を行う。

## 3.6 ロボカップ

### 3.6.1 ロボカップとは

ロボカップとは、「ワールドカップの優勝チームに人間のルールで勝つサッカーロボットを作る」ことを目標にしたグランドチャレンジである[3-4,5].1997年に名古屋で第一回の国際大会が開かれて以来毎年大会が開催され、2009年7月には第13回大会がオーストリアのグラーツで開催された。

ロボカップの発端は、北野らが日本の人工知能およびロボットの若手研究者有志に呼びかけて、1992年に開催したグランドチャレンジのワークショップに遡る[3-6].人工知能およびロボットの将来を考えて社会的に大きなインパクトを与えられる研究テーマを参加者が出し合って議論を行い、最終的にサッカーロボットをテーマに据えることに決めてロボカップはスタートした.最初にロボカップの構想を発表したのは1995年にモンリオールで開かれた人工知能合同国際会議(IJCAI)の場である.このとき、1997年に名古屋で開かれる次のIJCAIでロボカップの第一回が開かれることが決まった[3-6].

### 3.6.2 ロボカップにおける課題

ロボカップの主な課題としてあげられているのがサッカーである.サッカーをロボットに行わせるにはいくつかの課題がある.サッカーはリアルタイムのゲームであるため、瞬時の判断が必要となってくる.熟考して最適解を求めることよりも、瞬時に許容範囲内の解を求めることがサッカーでは重要となってくる.次に、サッカーは多数対多数で行うという課題がある.サッカーは11対11で行うゲームであるがゆえに、11のエージェントが協力して「敵より多くの得点を取る」という一つの目標を目指す.人間社会においても一人で問題解決をおこなうことより集団や家族で解決する事が多く協調システムの開発が課題となる.最後に、頭(ソフト的要素)だけではなく体(ハード的要素)が必要であるという点である.過去にあげられていた、人工知能の研究を例に挙げると、チェスなどは頭脳で行われるゲームであるが、サッカーはスポーツであるのでハード的部分も必要である.つまり、ロボットの知能化をハード的要素とソフト的要素から達せするという目標に対して、サッカーという課題は絶好の要素であると考えられる.

以上のようにサッカーはいくつかの研究課題として優れた特徴を持っている。そして、今後の重要になる知能化の技術開発の場として提案されている。具体的に課題として、

- ① 自律エージェント
- ② 実世界認識・行動
- ③ 実時間認識行動（リアルタイム制御）
- ④ 分散協調システム
- ⑤ 不完全・不正確・不確実情報環境での処理

の要素をサッカーは有している[3-6].

### 3.6.3 ロボカップの競技種類と発展

ロボカップリーグには、いくつかのリーグが存在する。最初に具体化したのはシュミレーションリーグである。シュミレーションリーグ(Fig.3-5)は、ロボットの実機を使うことなく、コンピュータ上の仮想フィールドで、それぞれ異なった人工知能プログラミングされた 11 対 11 のバーチャルロボットが 5 分ハーフのサッカーを行う。2次元、3次元フィールドでの競技のほか、コーチの技を競う競技もある。

実機リーグの方は、ロボットの大きさから小型リーグ(Fig.3-6)、中型リーグ(Fig.3-7)、四足ロボットリーグ(Fig.3-8)、ヒューマノイドリーグ(Fig.3-9)が存在する。小型リーグは、直径 18cm 以内のロボット 5 台 1 チームが、約 5.0m×3.4m の大きさのフィールド上で、オレンジ色のゴルフボールを使って試合をするリーグである。試合時間は 10 分ハーフで行われる。フィールド全体を見渡すカメラ、あるいはロボット搭載カメラからの視覚情報をベースに、ロボット同士がいかにシステムチックなチームプレイを構築していくかが見所である。一極集中型の管理体制で群知能行動を生成していく。

中型リーグは、30cm 以上 50cm 四方で高さ 80cm 以内のロボットが 5 台で 18 m × 12m のフィールドで、オレンジ色のボールを追う競技。ゴールは、2007 年度まで青と黄色のゴールを使用していたが、2007 年度以降は実際のサッカー同様に色なしのゴールと設定された。試合時間は 15 分ハーフ。多くのチームが 360 度見渡せるカメラを搭載、センサで自分とボールの位置をすばやく判断して動く。また、自律分散型の群行動を生成してロボットが行動する事が望まれる。

四足ロボットリーグは、ソニーの AIBO を 4 台 1 チームのサッカーリーグである。オレンジ色のボールを使い、フィールドは今年から 4 m × 6 m である。試合時間は 10 分ハーフ。共通プラットフォームを採用しているため、各チームのロボットプログラミングの優劣で勝敗が左右される。鼻の位置に取りつけられたカメラで、味方やボールの色を識別するために盛んに頭を振る。

ヒューマノイドリーグは、2002 年より正式種目となった自律型 2 足歩行ロボットのリーグ。PK や 2 対 2 での競技や、独自の機能を披露する「フリースタイル」競技が行われる。総合優勝者にはルイ・ヴィトン・ヒューマノイドカップが授与される。

また、ロボカップではサッカー以外を題材としたリーグも存在する。一つ目は、ロボカップサッカーで培われた技術を災害救助に利用というプロジェクトとしてレスキューリーグ(Fig.3-10)が存在する。レスキューリーグでは、地震などの大規模災害時を模擬して災害時の救助戦略を発展させようというシミュレーションと、現場に駆けつける自律型ロボットの開発を目的としている。実験・評価の場として競技会を活用し、国境なき救助ロボット隊の結成を目指し行われている。2 つ目は、家庭内でのロボットの行動を題材とした、ロボカップ@home リーグである。このリーグは、2006 年度ドイツ大会より開始された。内容は、人間とロボットのインタラクションをテーマとしており、ロボットが人間との共存をするために家庭内での様々な生活する上でのタスクをこなすリーグとなっている(Fig.3-11)。

最後に、ロボカップジュニアリーグ(Fig.3-12,13,14)である。ロボカップでは 2050 年に人間のワールドカップ優勝チームに勝つロボットチームを作ることを目指している。今の世代ではこの目標に対して実現は容易ではなく、次の世代に目標を託すことになる。それには将来の人材育成が重要な課題となってくる。そのためスタートしたのがジュニアリーグである。ジュニアによるロボットリーグは、物理、メカトロニクス、電子工学、プログラミングなどの総合的な分野を学ぶ分野として確立された[3-6,7,8,9,10,11,12,13,14,15,16,17]。

### 3.7 ロボカップ中型リーグ

ロボカップ中型リーグは、ロボカップサッカーリーグの中でもっとも大きなサイズの自律型移動ロボットによって行われるリーグである。中型リーグの特徴は、ハードウェアの開発とソフトウェアの開発双方を行わなければならない、ロボットは自律分散型で行動を決定しなければならないという点である。そこで、本研究ではロボットの知能化を検証するフィールドとしてロボカップ中型リーグを実験環境に設定した。3.2 節では、実験環境として選択した中型リーグがどのようなルールであるのかを説明する。

ロボカップ中型リーグでは、ロボット 1 台の大きさは、50 [cm] x 50 [cm] 四方の底面に入らなければならない。高さは、40~80 [cm] の大きさに入らなければならない (Fig. 3-15 参照)。ロボットは、自律行動を行う為のすべての要素が整っていないと、移動機構、コンピュータ、通信機能などすべてをこの大きさの範囲内に内包しなければならない。

コート大きさは、18 x 12 [m] の大きさでバレーボールのコートを一回り大きくしたサイズになる (Fig. 3-16 参照)。このコート内において各チームのロボット 5 台が自律的に移動することによってサッカーというタスクを行う。ロボット自体は、レフェリーボックスと呼ばれる審判が操作するゲームの進行コマンドを送信する外部 PC からの指令を無線 LAN によって受信し、行動を選択しなければならない。ロボットが行動を行う上で必要となる物体の認識を行わなければならない。中型リーグでは、画像データから物体を色によって認識する手法が主流である。実際には、2007 年度までは、ボールがオレンジ色、ゴールが青色と黄色、コートが緑色、ラインが白色となっていた。しかし、ルールの改正によりゴールの色は、廃止され現在は、ボールとフィールド・白線を認識して自己位置やボールを操作する方法が主流となっている。

試合時間は前後半 15 分で行われ、得点差によって勝敗がきまる。また、同点の場合は、トーナメント方式のときに限り 5 本ずつの PK 合戦が行われるが、PK 合戦も通常の試合と同様に自律的に行動しなければならない [3-18,19]。

## 3.8 モジュール構造を用いたプラットフォーム開発

自律型移動ロボットを設計・開発する際に、アミールの設計指針に示されるように様々な要求に応じた設計指針が必要となる(Fig.3-1,2). アミールによって示された設計指針によると, Fig.3-3, Fig.3-4 に示す設計指針は, ロボットに要求されるいくつかの項目を明らかにする事で示す事が出来る[3-22]. 以下にアミールによって示された設計指針に従って, 自律型サッカーロボットを設計する際に必要な項目を考える.

まず, 基本項目として, 設計するロボットの基本的な要求を考えると, 今回設計するロボットは, ロボカップ中型リーグ(MSL: Middle Size League)によって使用されるロボットである. そこで, Fig.3-4 に従い, 必要なタスク, 行動(Behavior), 使用する環境, 環境による拘束条件を考える. 以下に必要な条件を示す.

### (5) ロボットのタスク

- ・ ロボットが自律的にサッカーを行う

### (6) 必要な行動

- ・ フィールド上(18x12 緑色カーペット)での3自由度での行動
- ・ ロボットが指令コマンドを無線LANを通じてホストコンピュータから受信する事ができる
- ・ ロボットが, ボール(オレンジ色)を見つけて, ボールを運ぶ
- ・ ボールをキックする事が出来る
- ・ 白線を認識して自己位置を推定する事が出来る
- ・ 黒い物体を認識して障害物回避する事ができる

### (7) 使用する環境

- ・ フラットな平面のフィールド (18 x 12 [m]) Fig.3-16 参照
- ・ 屋内
- ・ 300 [lx]以上の照明環境
- ・ 人間と共存する環境

## (8) 環境による拘束条件

- ・ 2つのゴール（白色，位置が固定）
- ・ ボール（オレンジ色）
- ・ ライン（白線）
- ・ ロボットのサイズ（50 x 50 [cm]以下の底面 高さ 40 ~ 80 [cm]）

Fig.4-11 参照

- ・ ロボットの重さ（80 [kg]以下）
- ・ ロボットの色（黒）
- ・ 作動時間 15 分以上

以上の条件より，様々なロボット形態がある中で，開発するプラットフォームは車輪型のケーブルレスロボットを採用する(fig.3-17). 次にアミールの提案する手法を用いて，上位レベルの仕様を Fig.3-18 に示す[3-22].ここでは，タスクを遂行するために必要な行動とロボットが満たさなければならない仕様条件を考慮して必要な設計条件を示した. 次に，Figs. 3-19,20,21 に上位レベルの仕様から導き出される下位レベルの仕様を示す.これらの図において，赤色で示すのが，ハードウェアとして具体的に必要なものを示している.白色は，今回のプラットフォームを開発する上では重要ではない部分を示している.最後に，黄色で示しているのが，具体的に必要な機能やセンサ・行動を示している.この仕様から得られた全体の設計仕様を Fig.3-22 に示す.仕様が決定したので，仕様を満たすようにセンサを Fig.3-23 のように決定した.

ここでロボット全体の仕様が決定したので，次に過去に使用していたロボットの短所を考慮して，ロボットの設計における軸となる指針を考える. Fig.3-24 に示すロボットは，初号機のロボットで Fraunhofer AIS (Institute Autonomous Intelligent System)より購入したプラットフォームである[3-23].このロボットは，2つの駆動輪と2つのキャスタによって構成される車輪型移動ロボットである.カメラは，広角レンズ (70 [deg])を搭載した IEEE1394 カメラを使用した.カメラは，360 [deg]回転する事が出来るように，エンコーダ付 DC モータを回転させカメラ自体の角度を変更する機能を搭載している.ロボットは，障害物回避など動的な環境を移動する時に必要な IR センサや・距離センサ・タッチセンサなど4種類 11 個のセンサを搭載している.また，空気圧式のキック機構を踏査している.この初号機における機械的な問題点をいくつかあげる.

- (a) 車輪型の移動ロボットのため回転動作や並進方向への移動が難しい
- (b) 多くのセンサを搭載しているためデータの処理が複雑
- (c) ボールをトラッキングする為にカメラを回転させないのでハードウェア・ソフトウェア両面で複雑なシステムとなる
- (d) 人間と共存する環境で動作しなければならないのに安全性が保証できない
- (e) ロボカップという動的な環境上において信頼性が低い
- (f) 組み立て、メンテナンス、拡張性、トラブルシューティング、運送等の面を考えると機構が複雑すぎる
- (g) 他のチームのキック機構[3-25,26]が 6.0 [m/s]でシュート出来るのに対して、空気式のキック機構ではせいぜい 2.0 [m/s]である。

これらの問題点や設計指針をふまえて、全方位移動サッカーロボット“Musashi”をアミールのデザイン手法を用いて設計した。設計の指針として、柔軟な移動機構であり、信頼性やメンテナンス性、シンプル性、低価格を目標にロボットを設計する。

### 3.8.1 全方位移動ロボット “Musashi” の構成

Musashi は、3つのコンセプト導入し、アミールの手法に基づいて設計を行った。(1) は移動性能という考える上で重要なコンセプトとして取り上げる。次に(2) は、新たにアミールの手法を考える上で Principle 8 として追加項目として提案を行う。(3)は、Principle 7(3章を参考)に基づいた具体的な設計の方法を示す。

(1) Concept of omni-directional mobility

(2) Safety

(3) Modularity concept

#### 3.8.1.1 Concept of omni-directional mobility.

Musashi は、全方位移動機構を採用し、視覚情報に関してもオムニレンズを搭載した全方位型のロボットである(Fig.3-25).全方位移動機構を採用する事で、フィールド上での行動に対して高い操縦性を持つ事が出来る。つまり、Fig. 3-4 における移動性能(Mobile)を満たす為に使用される方針である。全方位移動機構の構造は、3D CAD(computer-aided design)である Autodesk Inventor を用いて設計を行った。ロボットは、同心円上に配置された3つのエンコーダ付きの 70 [W] DC モータを採用した。ギア比は、十分なトルクと加速度を得る為に 12:1 のものを選定した。車輪の速度は、モータドライバによって制御される。モータとの通信には RS232 通信を採用し、本体に搭載している PC から指令値を送っている。この移動機構によって、ロボットは最大速度 3.4[m/s]で移動する事が可能であり、初号機ロボットの問題点(a)を解消する事が出来る。

問題点(b)と(c)は、IEEE1394 デジタルカメラとオムニレンズを用いる事で全方位視覚を実現し解消した。Fig.3-25 に示すように地面からオムニレンズまでの距離  $H$  とカメラまでの距離  $h$  は、オムニレンズの死角を考慮して、死角にロボットが納まり、外界の情報をなるべく死角に入らないように設計した(Fig.3-27)[21]. オムニビジョンは、ボールを推定するだけでなく、外界の情報を取得する事で自己位置の推定や障害物回避等の行動を行う為の情報を全方位に取得する事が出来る為、センサの数を減らす事が出来る。

Table 3-1 に Musashi の諸元を示す。Musashi に使用されているセンサは、オムニカメラ、モータのエンコーダ、方位磁気センサの3つとなっており、初号機で用いていた 11 個のセンサから3種類のセンサに減らす事が出来た。Fig.3-24 に

Musashi の電源供給フローチャートを示す。メインの電源は、リチウムポリマー電池 (25.9 [V] 2000 [mAh]) を採用している。この電池より、カメラに必要な電源である 12.0 [V] や、マイクロコンピュータに必要な 5.0 [V] は、DC-DC コンバータを介して高圧する事で自作基板回路上において作成している。ロボットの消費電力を通常 40[W] 平均で使用すると概算すると、30 分間の駆動が可能である。

### 3.8.1.2 Concept of Safety

本節では、ロボットの安全性を考えた設計を行う。産業を目的としたロボットやサービスロボットなど多くのロボットが近年開発されている。産業用のロボットは、多くの自由度を必要とするマニピュレータ等で構成されている。この産業用ロボットは、制御を失敗すると、人間等に危害を与える事が考えられる。そこで、The international Organization of Standardization (ISO) は、一般的な産業用デザインの安全基準を ISO12100 によって提唱した[3-25]。さらに、ISO12100 をもとに産業ロボットの安全に関する ISO10218 が提唱された。ISO10218 では、オペレータの安全性を守る為の産業用ロボットの安全基準を提唱した。このように産業界のロボットの安全な設計指針が提案され、実際に Fig.3-26 に示すように産業用ロボットには様々な安全方策が施されている。

このように技術者は、社会の要求を満たしながら、ロボットの安全性を設計上確保しなければならない。ロボカップ中型リーグの場合も、ロボットと人間が共存する環境の中でロボットは、動かなければならず人間の安全に危害を犯しかねない。それ故に、ロボカップのロボットにおいても産業ロボットと同様に安全を考えなければならない。

ロボカップの最終目標の一つは、人間のワールドカップのチャンピオンに勝つ事である。しかしながら、現在のロボカップ中型リーグのルール上では、人間との試合を想定した安全方策はなされていない。そこで、ロボットの安全に関してリスクアセスメントを記した ISO14212 [3-26,27]を用いてリスクアセスメントを行う事で安全方策を行う。ここで、ロボットの安全な設計の手順を考えると Fig. 3-27 のように設計する必要がある。この安全方策をアミールが提案する Fig. 3-3 に示すデザイン設計に取り込む(Fig. 3-28 参照)。そこで、ロボットのハードウェアの知能化として設計コンセプトの Principle 8 として本節では、全方位サッカー移動ロボットを題材として移動ロボットの安全設計指針を示す。ここでロボットの安全の定義を示すと、ISO の基準より、安全とは、“受け入れが

たいリスクがない事”を示す。以下に、提案する設計上の原則を Musashi の安全設計を例にして提案する。

## Design Principle 8: The principle of the safety mobile robot

安全なロボットを設計する為には、Fig.3-27 における設計手順を踏む必要がある。はじめに、ロボットは、制限範囲を決定しなければならない。

### (1) 行動範囲の決定

今回使用するロボットでは、用途は、競技会における試合やイベントによるデモンストレーションと設定する事が出来る。また、その仕様場所は、屋内環境であり Fig.3-16 に示したフィールド内での環境と限られている。

### (2) 危険源の同定

第 2 段階では、危険減の同定を行う。今回の使用するロボットでは、以下のような危険源が存在する(Fig. 3-29 参照)。

- ・ 緊急停止ボタンの不装備 (暴走時に危険)
- ・ 地面と底面部分にギャップがある (足を引かれる, 押しつぶし)
- ・ 鋭利部分との接触 (接触時に皮膚を切る)
- ・ ロボットの素材がすべて金属

このように、危険源を特定して、危険源に対するリスクの見積もりを行わなければならない。

### (4) リスクの見積もり・評価

リスクとは、JIS の規格により『潜在する危害のひどさ』×『発生確立』によってリスクを見積もる事が出来る。Fig.3-30 に示すようにリスクを見積もる際は、ツリー形式によって見積もる事が必要である。この時、クラス I は、受け入れ可能なリスク、クラス II は、許容可能なリスク、クラス III は、受け入れられないリスクと定義する。この定義に従ってリスクを見積もる。具体的に例を示すと小川らによって本ロボットのリスクの見積もりを衝撃に関して行った例がある

[3-32]. このように危険源に対してロボットのリスクを見積もる必要がある。実際に Musashi の危険である部分に対して、リスクの見積もりをした例を Table 3-2 に示す。

#### (5) リスクは適切に低減されたか

ここで、リスクを評価してリスクが残っている場合、設計上に問題があるため安全な防護策を取る。本手法では、ISO12100 を参考として、リスクの低減方法を 3 ステップメソッドを利用して提言する手法を提案する。そこで、具体的に全方位移動サッカーロボット”Musashi”を例として、リスクの低減方法を示す。

#### Step 1.

最初の段階は、本質的に設計段階で安全性を高める方法である。つまり、ロボットが危害を加える可能性があるものは、すべて設計の段階で取りのぞくべきであるという考えである。しかし、多くのロボカップ中型リーグで使用されているロボットは、設計・開発が終了しており、この段階で危険な部分を取り除く事が難しい。そこで、次のステップに進む。

Musashi ロボットの場合、設計段階での安全上の指針は、いくつかの部分において考えられている。例えば、バッテリーに関してである。バッテリーは、ロボットが動く為には重要な要素の一つであり、必要不可欠なものである。Musashi では Table 3-1 で示すようにリチウムポリマーバッテリー(3.6 [V] 2000 [mAh])を 7 セル使用している。一つ一つのセルは、3.6 [V]で最大 4.2[V]まで充電可能であり、3.1 [V]まで使用する事が可能である。つまり、満充電の場合バッテリーは、定常電圧 25.2[V]を示し、最大 29.4 [V]、最低で 21.7 [V]となる。バッテリーの安全方策として、最低電圧より電圧が下がったり、最大容量を超えたりする事を避けるように内部で電圧保護回路やブレーカなどを設置している。また、バッテリーをロボットに搭載しない事は不可能なので、もしバッテリーが火災などを起こした場合も 70 [%]以上を設計上カバーで覆うようにしている為、本体自体の火災や外部に影響を与えないようにしている。

## Step 2

第2段階では、安全方策として、安全防護方策として安全装置の設置や、追加で防護策をとるなどを行う。例えば、緊急停止ボタン等を設置するなどの対策があげられる。しかし、緊急停止ボタンの不具合等を考えると、人間が中型リーグでは介在しているので、レフリーや参加者などに危険性が残る。

Table 3-2 に示すように、いくつかのロボットには設計段階では取り除けなかった危険減が存在する。例えば、底面の部分のエッジの部分の部分が鋭利である。このような部分に対しては、緊急で停止できる措置やスポンジなどの柔らかい素材で鋭利な部分をカバーする等の対策を行う事が必要である。

## Step 3

最後の段階は、ユーザに対して、使用上の情報提供を行う事である。リスクを第1、2段階で取り除けない場合は、ロボットの危険性をユーザに対して提示する事で、危険を取り除くという方策である。実際に例を挙げると、中型リーグの場合、審判がロボットの動作している状況に存在している。つまり、審判はロボットの危険性を**知る**必要がある。また、審判自体がロボットの危険性を把握し、安全な靴や服装を心掛けるなどの対策をとる必要がある。

このように3つの段階を踏んでロボットのリスクを提言させる。このように安全を設計指針の段階から考慮する事でロボットが人とのインタラクションを行うような場面に遭遇しても、安全性が保証される。よって、Fig. 3-27で示したデザイン原則によって、ロボットのリスクを設計段階でリスクアセスメントを行い、安全方策を施す事で安全なロボットを設計する事が出来る。

### 3.8.1.3 Concept of modularity

#### *Step 1: Description of the Robot System Architecture*

本節では、ロボットにモジュール構造を用いた設計指針について述べる。はじめにモジュール構造を考える上で、ロボットのシステム構造を簡単に述べる必要がある。Fig. 3-31に示すように”Musashi”は、ノートPCを搭載している。ノートPC上では、画像処理や制御、通信やデータの受け渡し等を行っている。スタートやストップ、コーナキック等の外部PCであるレフェリーボックスPCからのコマンドを無線LANでフィールドの外から受けてロボットは行動する。アクチュエータとしては、DCモータを3台搭載し、オムニホイールを用いて移動し、エンコーダによってそれらの回転数をモータドライバを介してPCへと送信している。単純なシステムを実現するために通信プロトコルは、IEEE1394(全方位視覚センサ)とRS232(アクチュエータ)の2系統とした。モータドライバとPCとの接続はUSBシリアルコンバータを介しておこない、PCはUSB通信からシリアルコンバータを介してモータドライバに回転数を送信してロボットを制御している。

データの流れに関しては、PCはIEEE1394接続によって全方位センサから画像データを取得し、PCにおいて画像処理を行いモータへの指令値を決定する。モータへの指令値は、RS232通信プロトコルを用いてモータドライバへと送信される。モータドライバは、送られてきた指令値をもとに速度制御を行い指令電圧をモータに送信する。キック装置も同様に視覚センサからの上方をもとにキック指令がRS232通信プロトコルを用いてマイクロコンピュータに送信される。

## Step 2: Definition of the Robot System Architecture

Fig. 3-31 で示すロボットの構造をもとにして、ロボットの各パーツに対してシングルモジュール(SM)を決定する。シングルモジュールを決定する時に我々は、“同じようなハードウェア上の性質を持っているか、同じようなインターフェースを持っている”という事を考え、グループ分けを行った。Fig. 3-32 で示すように破線で示している部分を7つに分けて SM を決定した。例えば、USB ハブと4つの USB/RS232 コンバータは、共通のインターフェースを所有している為、USB モジュールとして SM として定義した。他の例をあげると、オムニホールとモータは機械的にもつながりが強くハードウェア上同じ性質を持っているため(MW modules)として定義した。ここで、明記しておかなければならないのは、デザインコンセプトとして、SM のデザインは、インターフェースとして、バックプレーンとなる基板回路に直接つながるという事が可能でなければならないという点である。コネクタや固定具を含むインターフェースは、固定して動作している間にかかる外力を考慮しておかなければならず、その上、交換しやすいように設計しなければならない。Fig. 3-33 に示すように USB モジュールは、インターフェースとして、16Pin のコネクタをバックプレーンとのインターフェースとして設置しており、その他にバックプレーンとピンによって振動しないように固定具で固定されている。そのため、振動等によってコネクタが外れてしまう事を防いでいる。また、SM は一つ一つが独立に機能を持っているため故障の早期発見につながり、また固定具の交換も容易である。

もう一つの SM を決める上でのコンセプトは、安全性である。例えば、バッテリーモジュールは、安全性を考慮した上でモジュール化を行っている。Musashi は、リチウムポリマー電池を使用している。そこで、従来ロボットに使用される電池よりも爆発の危険性が高い。バッテリーモジュールは、上部と底部によって構成されている(Figs. 3-34)。このパーツは、バッテリーと配線部分を覆っており、約全体の 60%をモジュールのパーツによって覆っている。また、バッテリーモジュールはバッテリーボックスに装着する為、装着時には約 80%以上が直接的に外にむき出しになっていない。そこで、電池は、爆発をした際にもほとんどの部分がモジュールを構成するパーツで覆われており爆発による外部への被害をもたらす可能性が従来よりも低くなっている。

### *Step 3: Definition of the merged module*

モジュール化の概念は、ワイヤーの数を減少させる努力とSMを合併することによって3段階目が行われる。合併したモジュール(Merged Module: MM)は、フローチャートをもとにSMの接続と考える組み合わせられ、「バックプレーン」に接続するという概念によって実現した。バックプレーンは、MMのSMとの通信経路基板と考えることができる。たとえば、USB、キッカーモジュール (KC) とモータドライバ (MD) モジュールとを接続する事が可能である。バックプレーンは、5つのSM (USB, KCと3つのMDモジュール) を接続して、複雑な配線接続の問題を解決した。この5つのSMを接続したMM(バックプレーンを含む)は、中心制御モジュール (Fig. 3-35) と呼ぶ。

Musashi のモジュール構造は、最終的に2つの主要な MM にまとめる事ができる (下部モジュールと上部モジュール(Fig. 3-36))。このモジュール構造を用いる事でメンテナンス性、トラブルシューティングと輸送の容易さを考慮する事ができ、従来のロボットよりも格段に性能が向上している。このモジュール化によって、従来のロボットの問題である(e)と(f)を解決することができた。

### 3.8.2 全方位置動ロボット”Musashi”のキック機構

ロボカップの歴史をたどると、1997年の中型リーグ設立当初、開発されたロボットには、キック機構は存在しなかった。1998年になると、ドイツのフライブルク大学によって初めてのキック機構が開発され、フライブルクのチームはキック機構を用いる事で優勝を勝ち取った[3-4]。1998年以降、キック機構の開発は盛んに行われるようになり、キック機構は勝敗を左右する重要な機構の一つとなった。

今回、Musashiに搭載するキック機構は、ねじりバネを用いた機械式のキック機構である。キック機構の設計指針としては以下の2つがあげられる。

- (a) ハイスピードのシュートをうつ事が出来る  
(おおよそ 5.0 – 6.0 m/s 以上)
- (b) ボールを宙に浮かす事が出来る (高さ 1.0 m 以上)

一般的に、3つの機構がバネのエネルギーを使う為には必要である。

- (1) バネを縮めてためるメカニズム
- (2) バネを一定の力をためたまま保持する為の機構
- (3) バネの力を解放する機構

新しいキック機構を3つの機構を開発する代わりに本手法では、カムを用いたチャージ機構を用いた。カムチャージ機構は、タッチセンサにカムを一つ使いカムの回転角度によって制御を行っている。楕円形上のカムを回転させ、そのカムがタッチセンサの ON/OFF の制御を行っている。タッチセンサが ON の時はもう一つのカム（勾玉上のカム）が回転し、バネを縮めてエネルギーを蓄える。その後、タッチセンサが OFF になったときにその地点で勾玉上のカムの動作を停止させ、バネの力を保持して止める。その後、楕円状のカムが回転すると力を放出する。この機構の詳細は、“JumpingJoe”に用いているメカニズムを参考に開発を行った[3-29,30,31](Fig. 3-36)。

この機構を用いる事で、強力なキック機構を開発する事に成功し、問題点にあげた(g)を克服する事が出来た。

### 3.8.3 まとめ

本節では、全方位置動サッカーロボット“Musashi”の開発をアミールによって提案された”Concept of Intelligent Mechanical Design for Autonomous Mobile Robot”に基づいて開発を行った。Musashi は、(a) Omni-directional Concept, (b) Safety Concept, (c) Modularity Concept の3つの主要な設計指針をもとに開発を行った。

Omni-directional Concept を用いる事で、全方位置動機構のプラットフォームを開発し、以前のロボットでは、運動学的に非常に困難であった横方向や回転方向への運動を実現し、操作性や運動能力の向上に成功した。また、全方位カメラを採用する事で、ロボットの視野を拡大し、カメラでの物体の同定能力や環境認識能力を向上させる事に成功した。また、以前使用していたプラットフォームに比べ、11個あったセンサを4つに減少させる事に成功した。

次に、様々な移動ロボットが現在開発されているが、安全性を考慮した設計はほとんどのロボットでなされていない。そこで、本研究では、ISOを基準とした産業用ロボットの安全性を考慮した安全なロボットの設計を行った。その際にロボットのリスクを見積もり、どのようにそのリスクを回避して行くかの指針を示した。このアプローチは、アミールによって提案されたハードウェア的なロボットの知能化の観点から考えても重要な要求の一つである。よって、ロボットの知能化を果たす為、安全性に関して設計を行い、リスクアセスメントを行う事でロボット危険部位を明らかにするという手法を設計原則 Principle 8として提案した。また、安全な設計、保護方策、情報開示という3つのステップを用いた保護方策について提案した。

最後に、ロボットの各パーツをモジュール化する事によって、ロボットのメンテナンス性や信頼性の向上、輸送の際の便利性の向上などを実現した。具体的には、3つのステップを用いる事によってロボットの各パーツをモジュール化して、モジュール化したパーツをさらにモジュール化する事によってロボットのシステムを大きなモジュール構造として開発した。

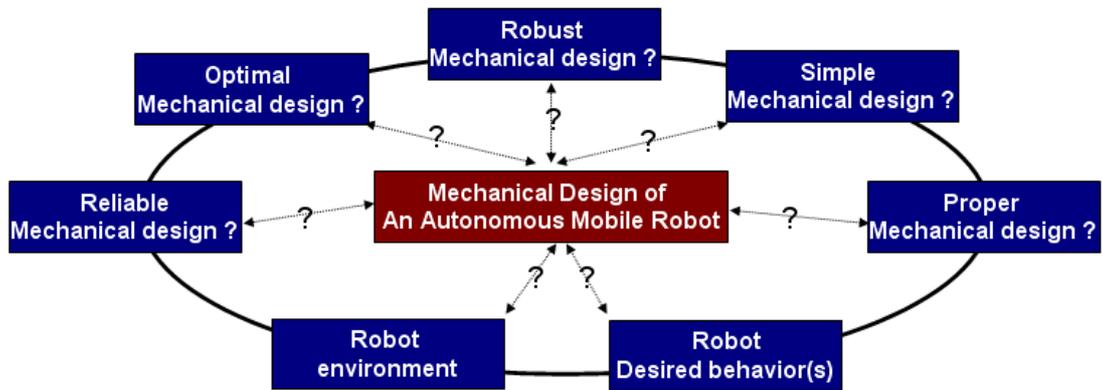


Fig. 3-1 Mechanical design of an autonomous mobile robot and related factors [3]

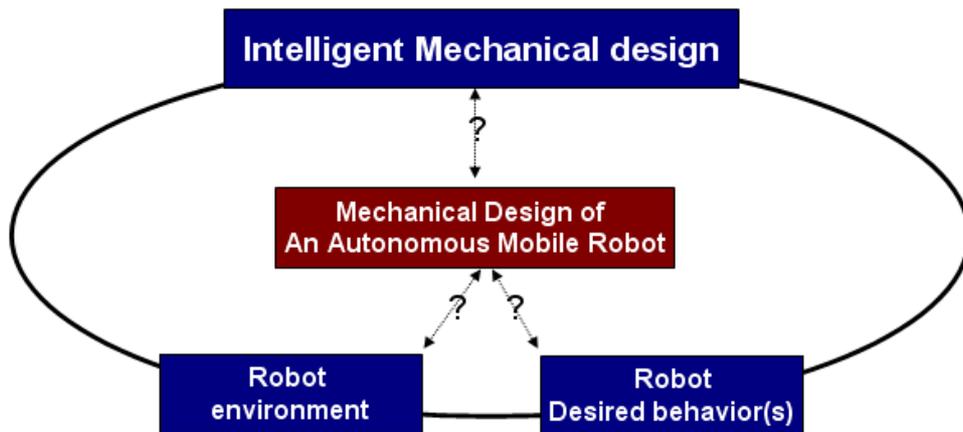


Fig. 3-2 New approach to mechanical design of an autonomous mobile robot and related factors [3]

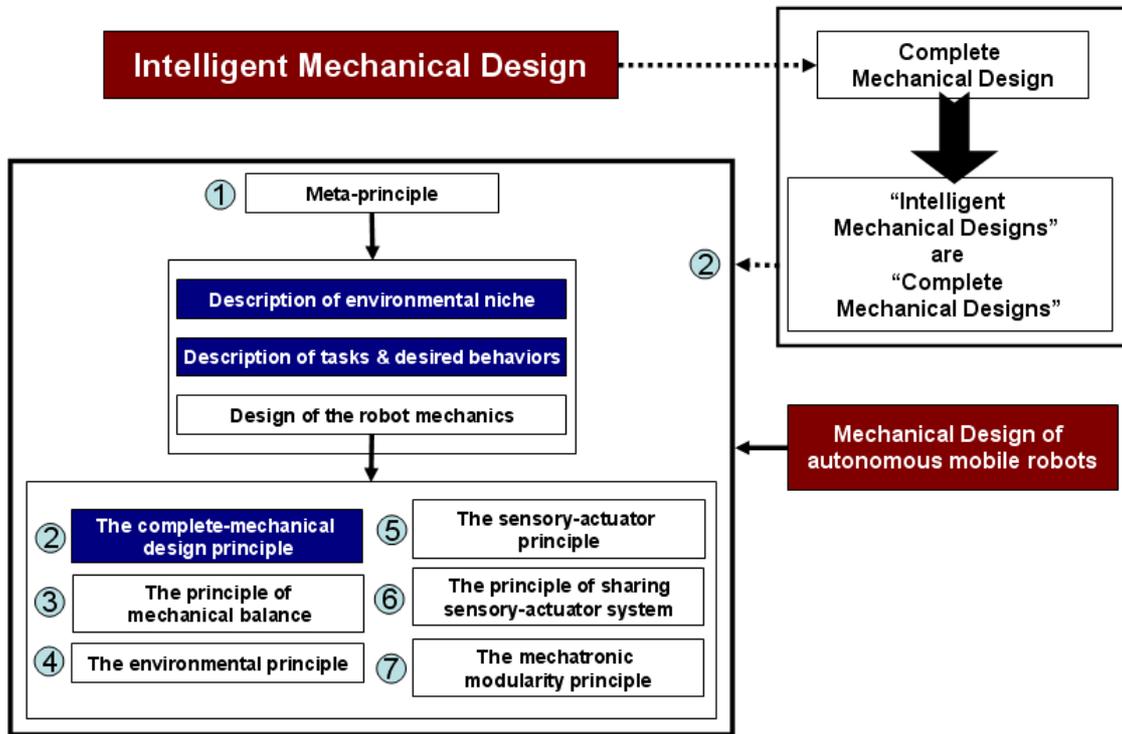


Fig. 3-3 Overview of design principles of mechanical design of autonomous mobile robots [3]

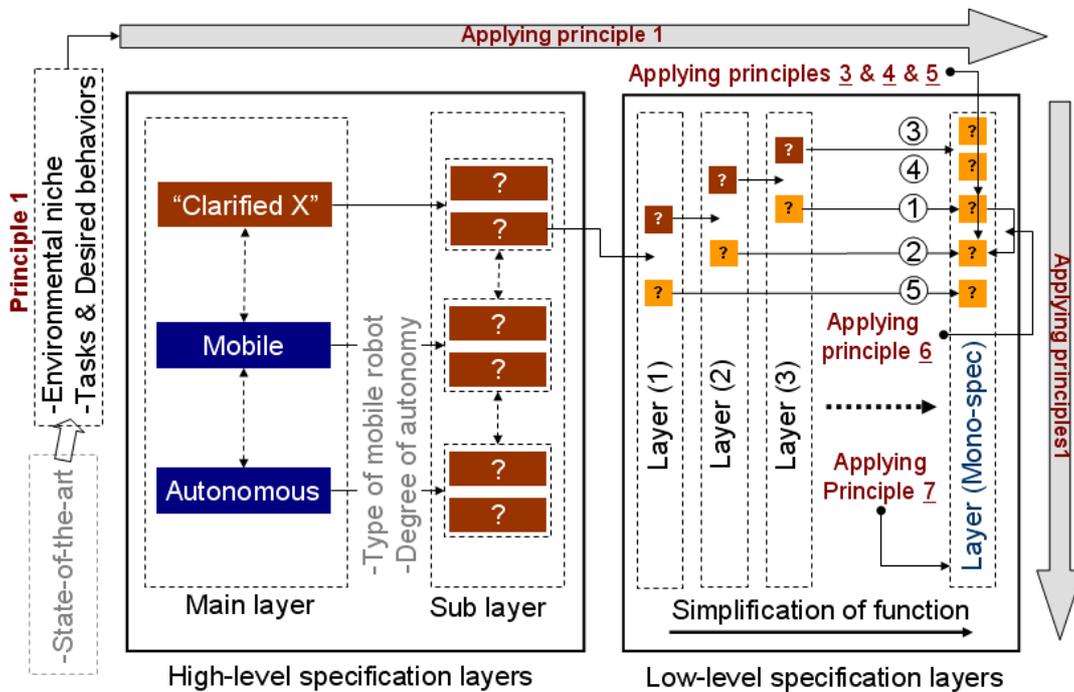


Fig.3-4 Overview of “Mecha-telligence methodology” for mechanical design of autonomous mobile robots with defined task “X” [3]



Fig. 3-5 Simulation league

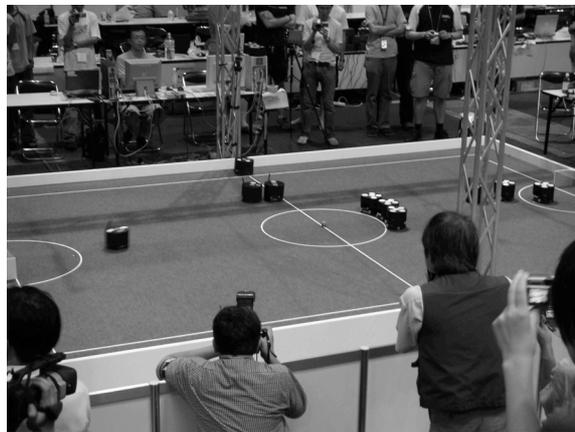


Fig. 3-6 Small size league



Fig. 3-7 Middle size League

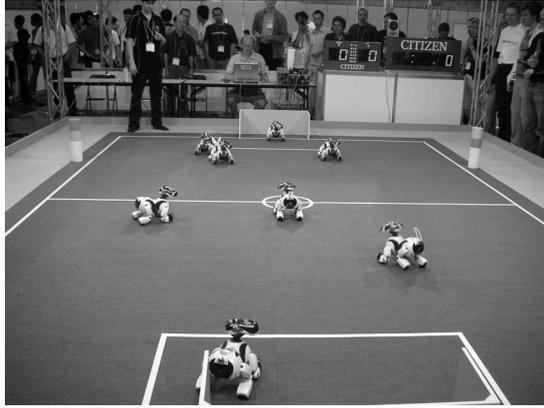


Fig. 3-8 4-leg league



Fig. 3-9 Humanoid league



Fig. 3-10 Rescue league



Fig. 3-11 @home league



Fig. 3-12 RoboCup Jr dance league



Fig.3-13 Robocup Jr. soccer league



Fig. 3-14 RoboCup Jr. rescue league

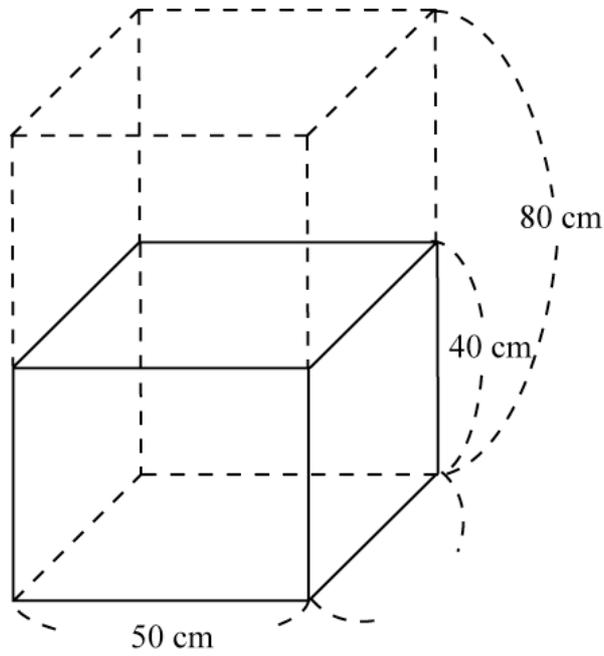


Fig. 3-15 Limitation of a MSL robot size

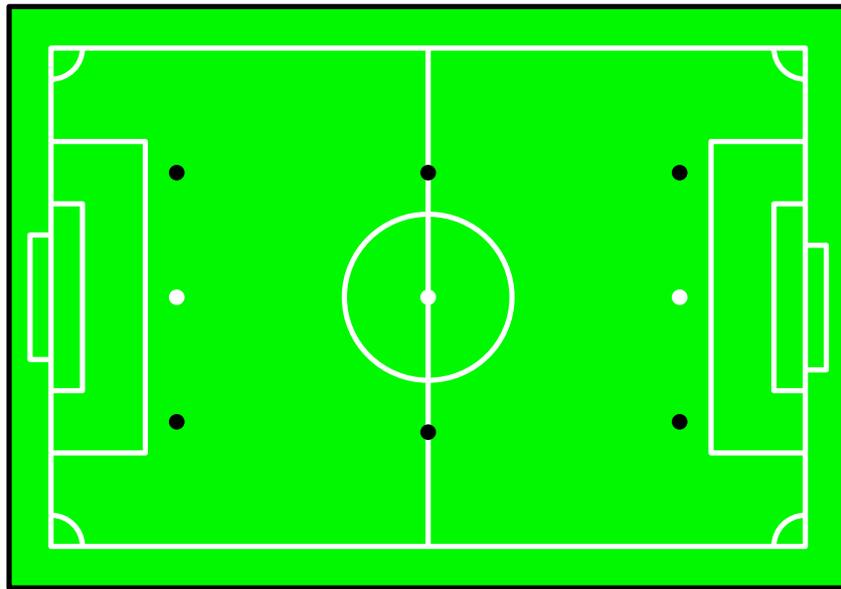


Fig. 3-16 The field of RoboCup MSL (18 x 12 [m])

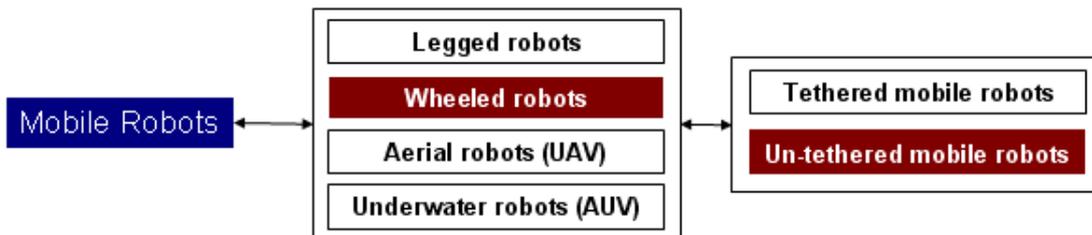


Fig. 3-17 The selected type of soccer robot [19]

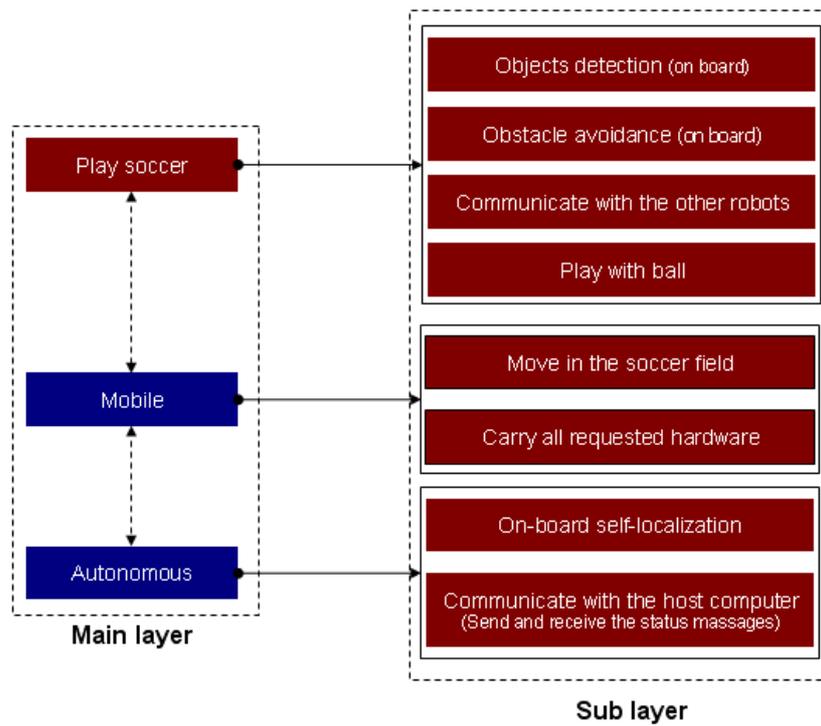


Fig. 3-18 The high-level specification layers including main-layer and sub layer [19]

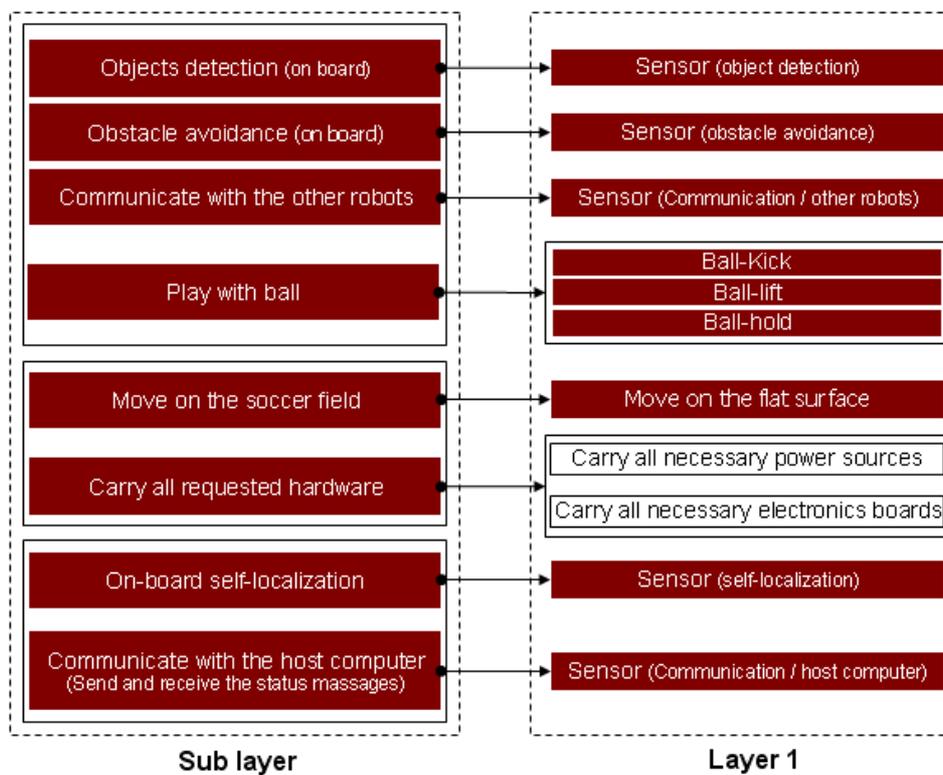


Fig. 3-19 Layer 1 generated from the sub-layer[19]

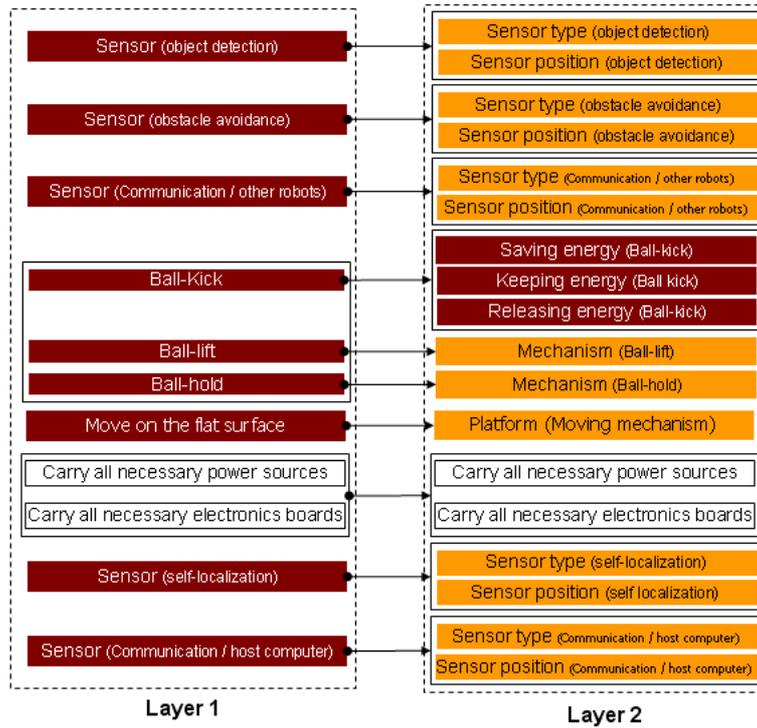


Fig. 3-20 Layer 2 generated from the layer 1 [19]

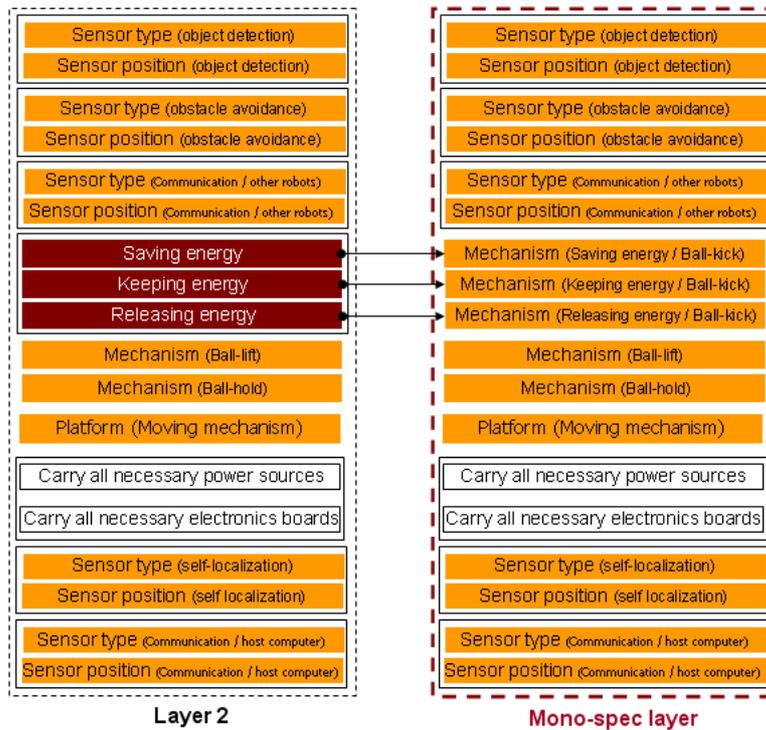


Fig. 3-21 Mono-spec layer generated from the layer 2 [19]

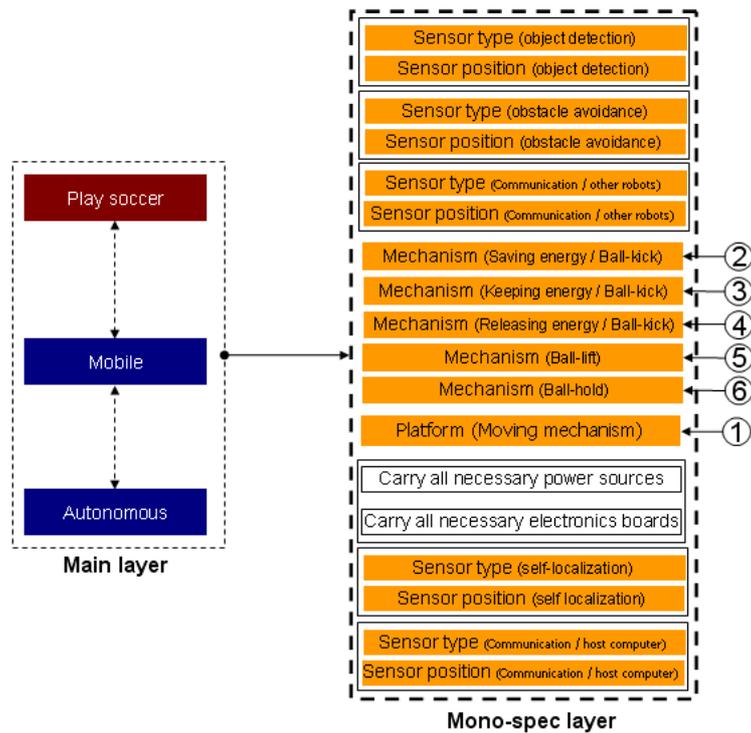


Fig.3-22 Results of the mono-spec layer generation and the function design priority [19]

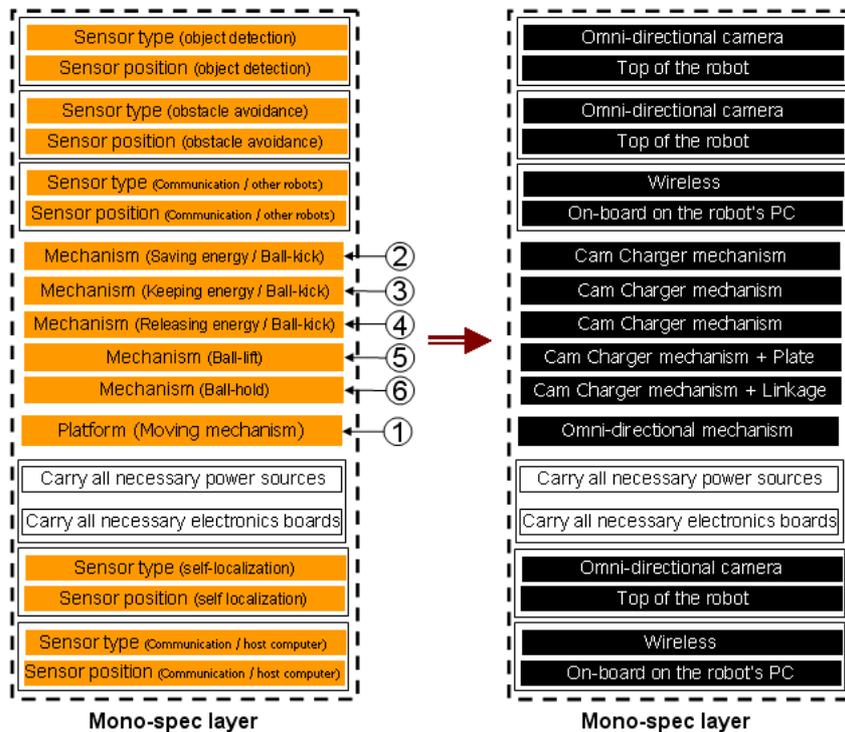


Fig. 3-23 The results of design [19]

The black color shows our solutions for the listed problems in the mono-spec. layer

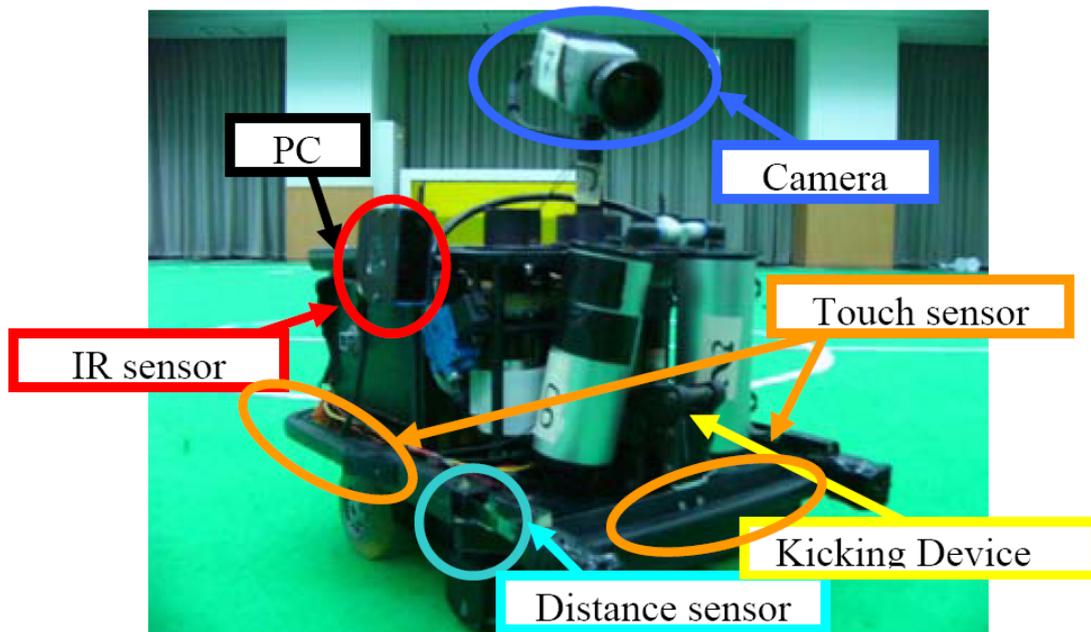


Fig. 3-24 The first version of our robot equipped with eleven sensors

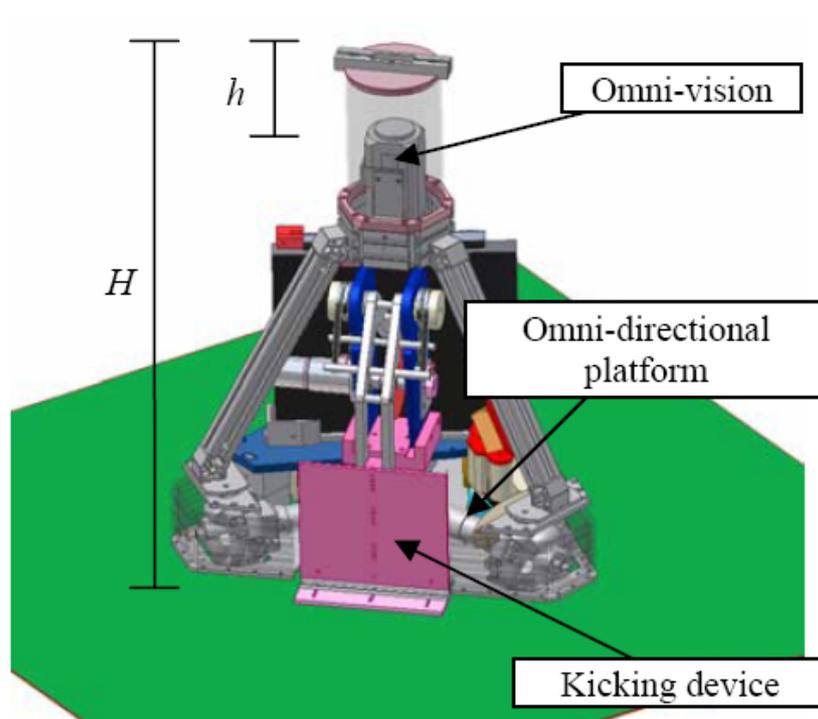


Fig. 3-25 Musashi robot platform

“Musashi” robot includes an omni-directional platform, an omni-vision, and a strong novel ball-kicking device, designed by Autodesk Inventor 3D-CAD

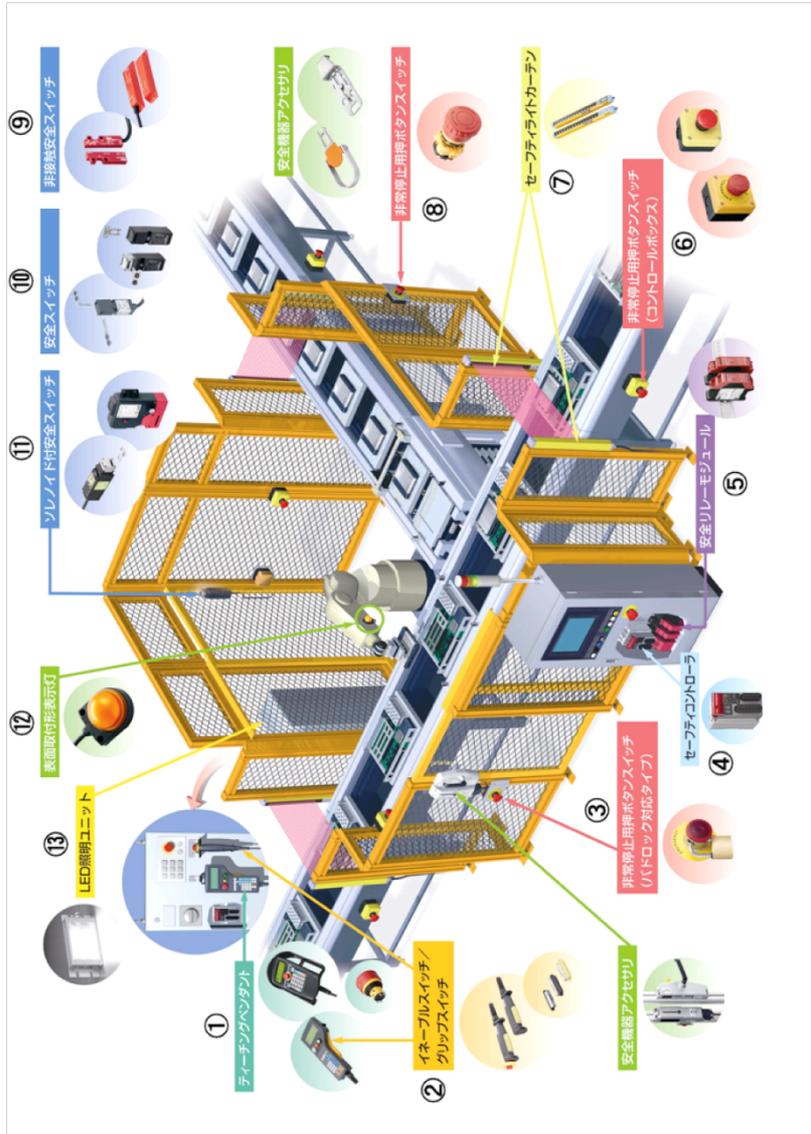


Fig.3-26 The safety example of the industrial robot (IDEC Co. Ltd. )

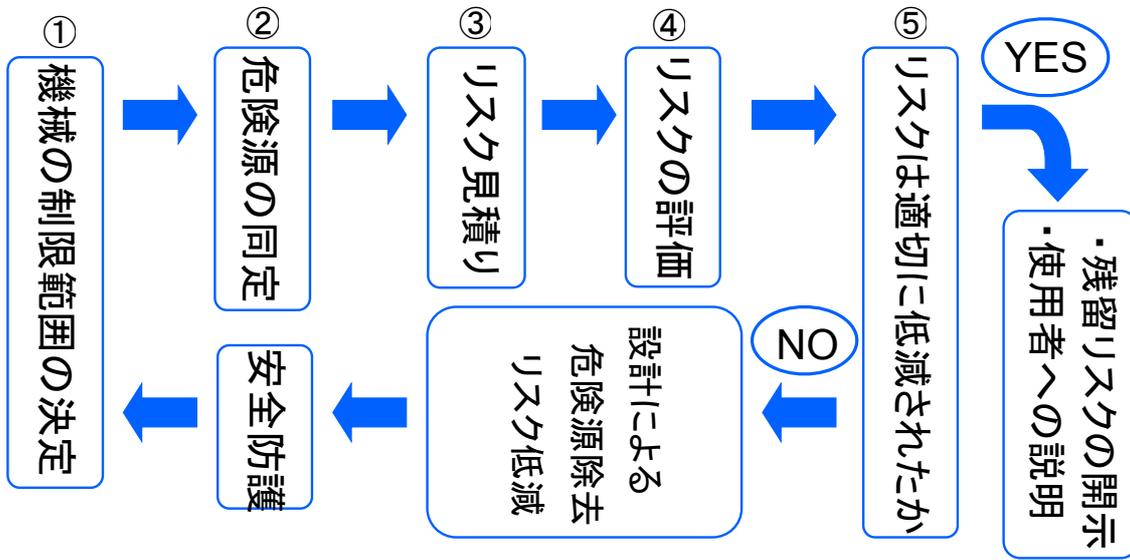


Fig. 3-27 The methodology of the safety robot design

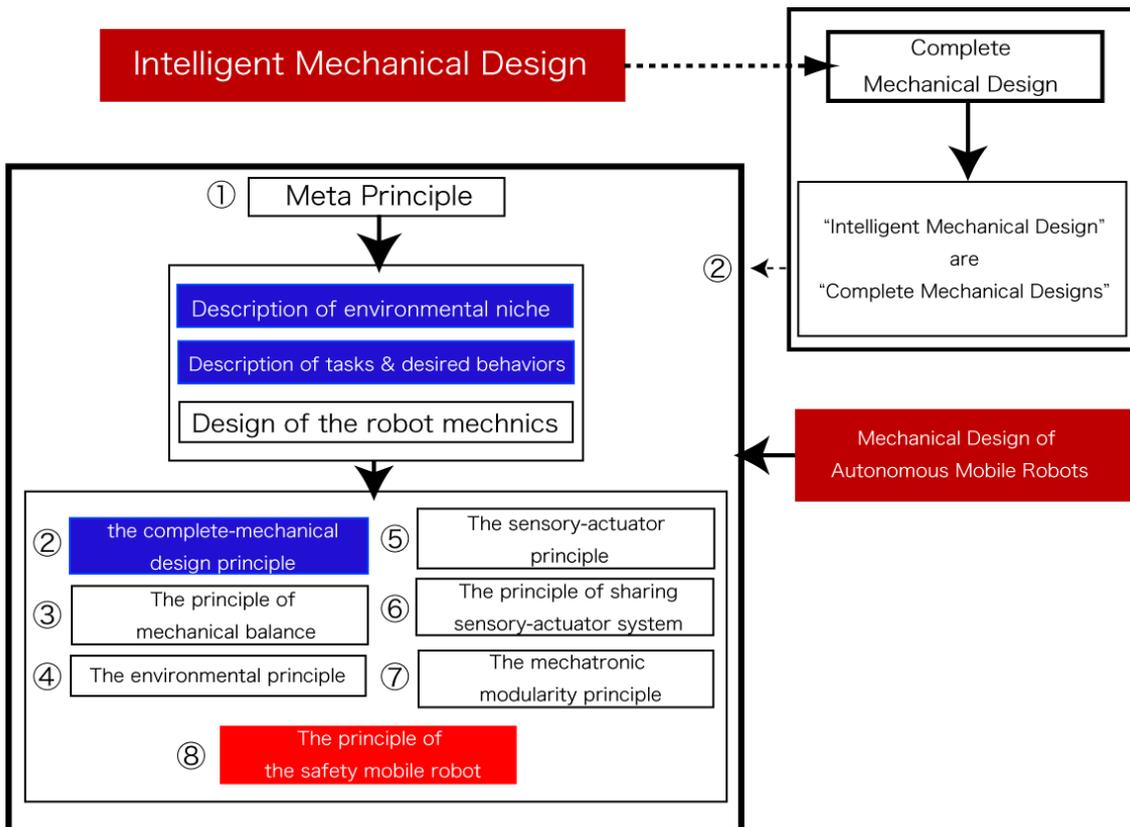


Fig. 3-28 New Concept of Intelligent Mechanical Design for Autonomous Mobile Robot

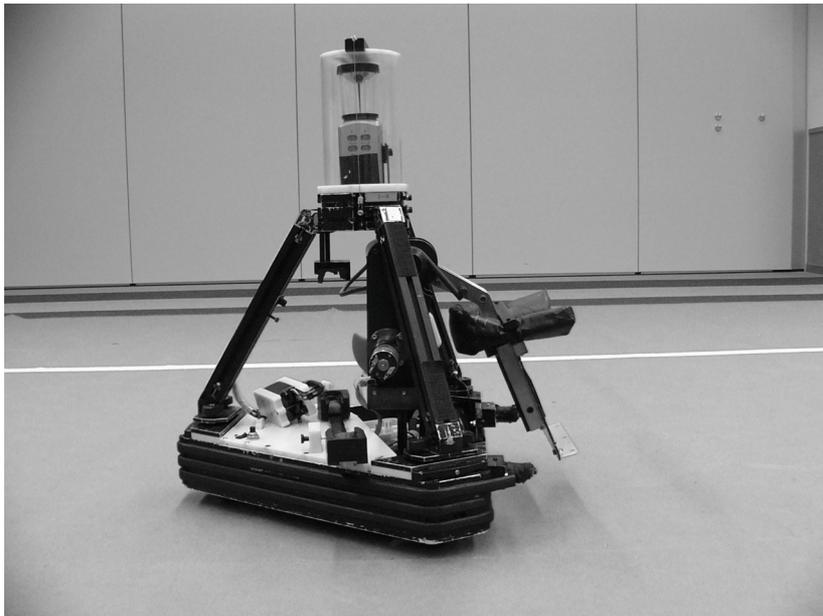


Fig. 3-29 Identification of the danger point

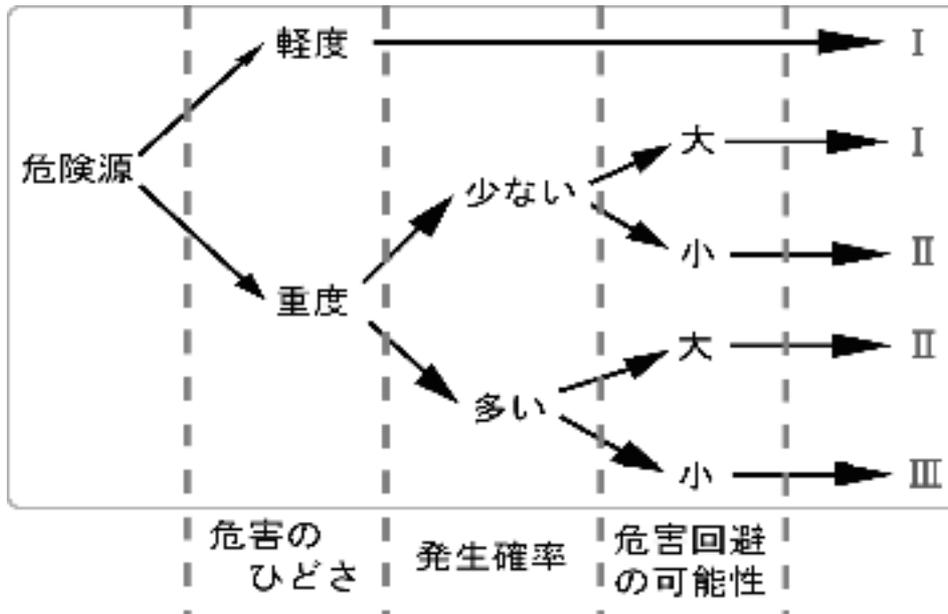


Fig. 3-30 Estimation of the risk

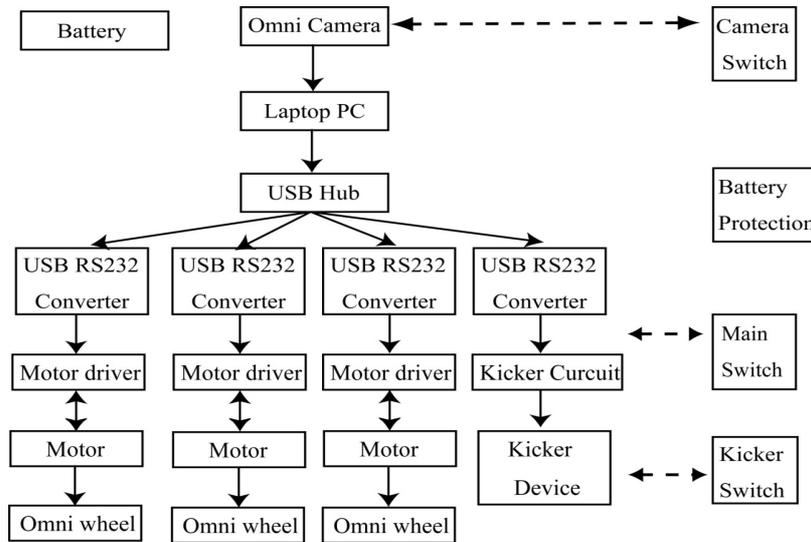


Fig. 3-31 Flowchart of the Musashi robot

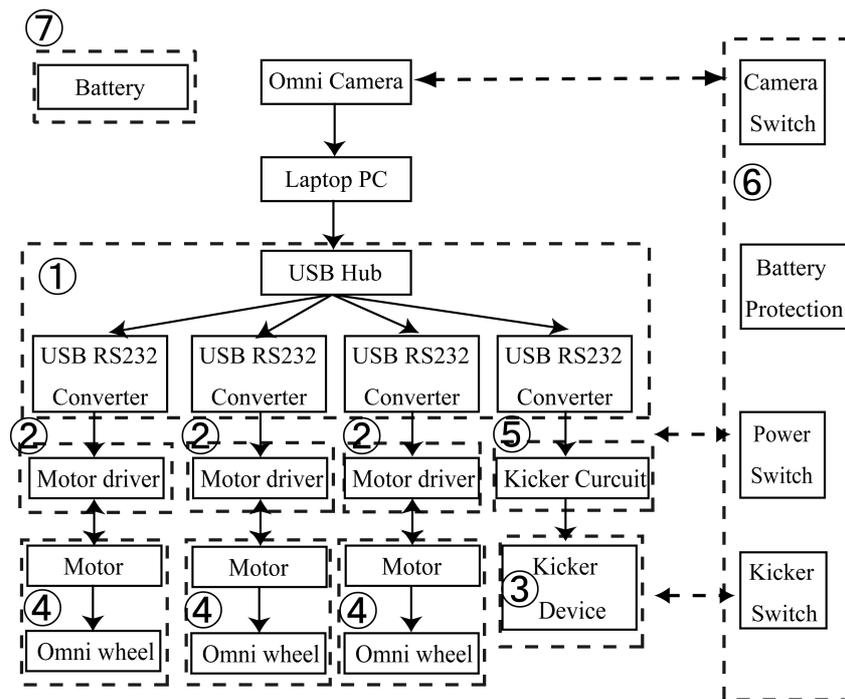


Fig. 3-32 Flowchart of the Musashi robot:

Each dash square area describes basic modules. Area 1 indicates the USB module, are 2 the motor driver (MD) module, area 3 the kicking device (KD) module, are 4 the motor and wheel (MW) module, area 5 the kicker circuit (KC) module, are 6 the main, power and camera switches (SW) module and are 7 battery module.

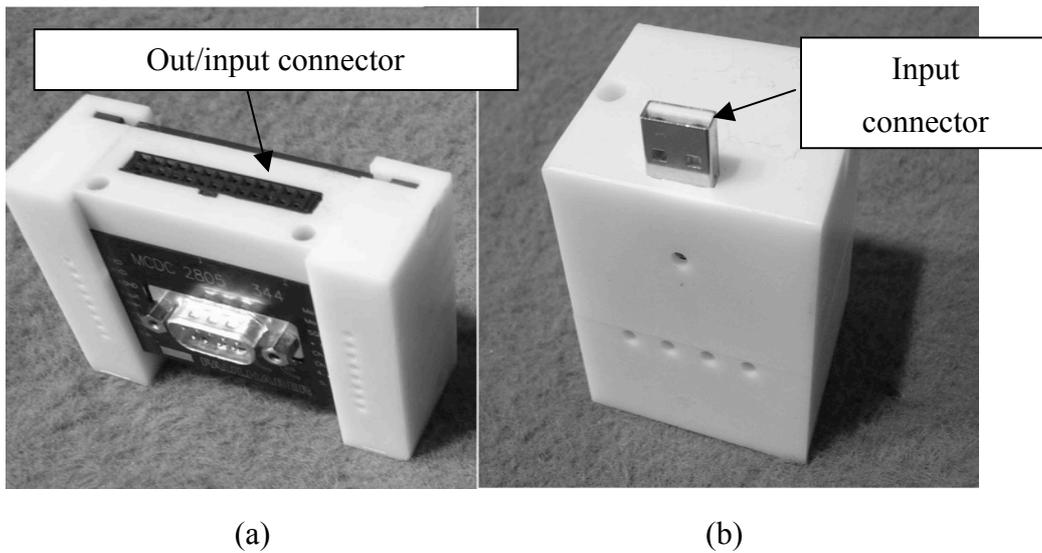
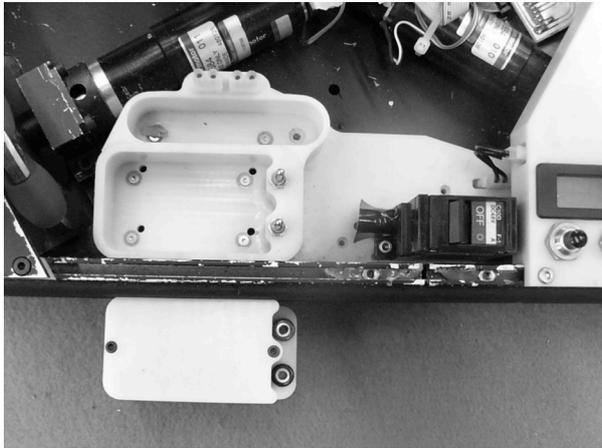


Fig.3-33 Single modules

(a): Motor driver module (MD module) including a connector as I/O port (b): USB module consists of two connectors for input and output port. Output connector is at the bottom surface.



Figs. 3-34 Battery module

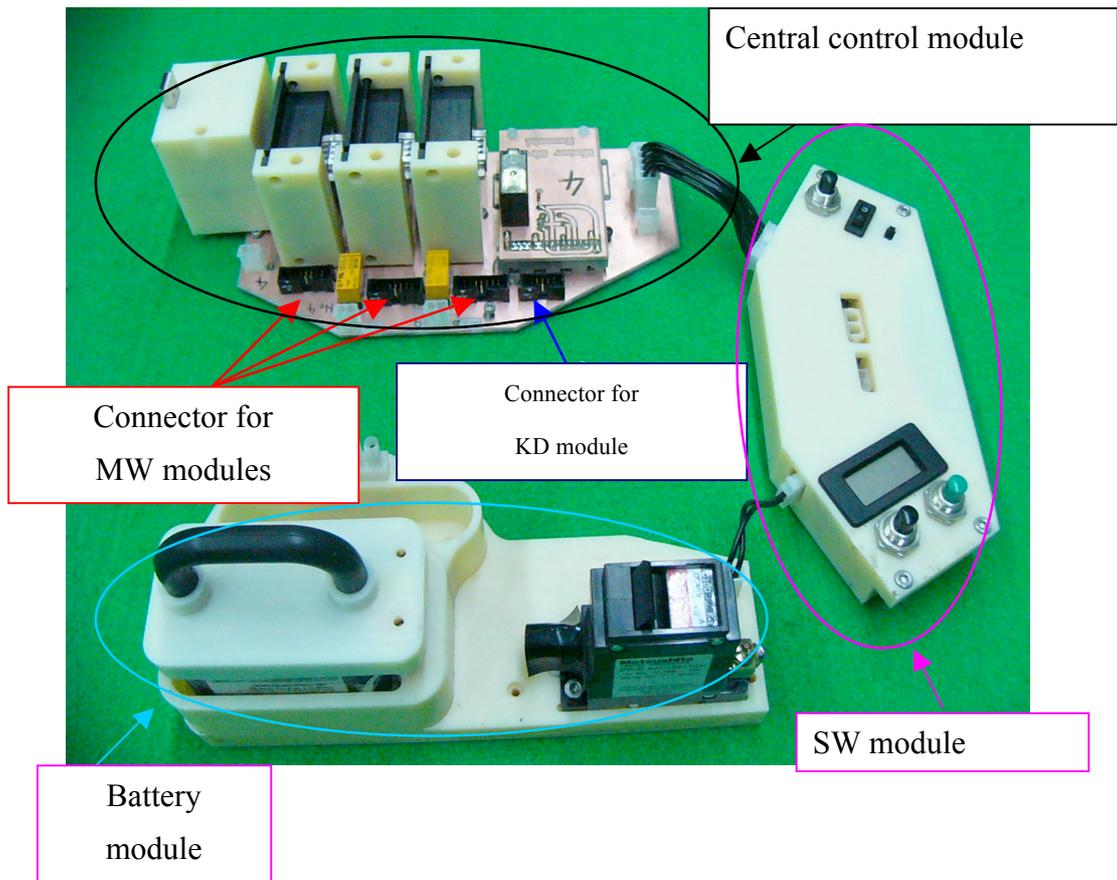


Fig.3-35 Modules of “Musashi” robot hardware

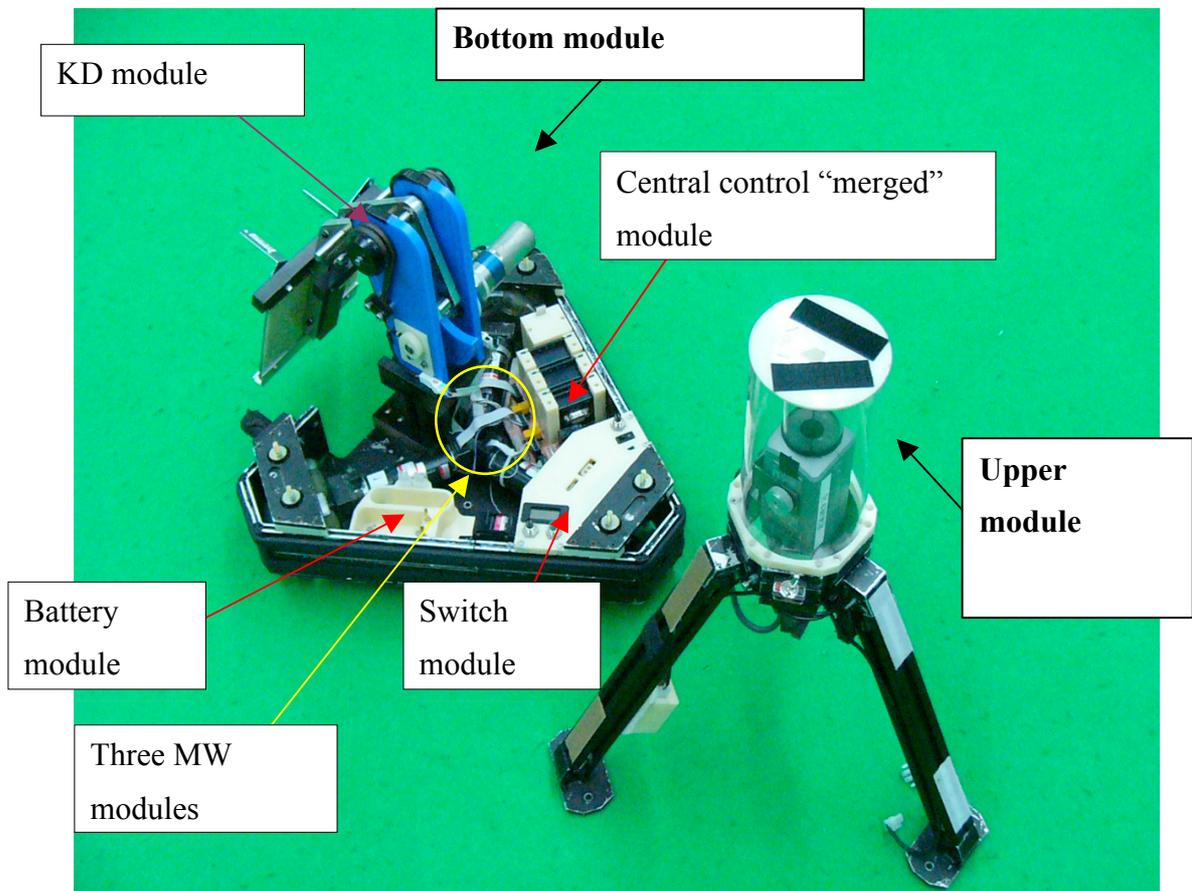


Fig.3-36 Modules of "Musashi" robot

Table 3-1 Specification of a “Musashi” robot

Item	Specifications
Size	500 × 500 × 800 mm
Total Weight	18.0 kg
Actuator	DC-motor × 3 (Maxon, 24.0 V, 70 W) Motor Driver × 3 (Faulhaber, MCDC 2805)
Power Supply	Li-polymer battery (4.7 V × 7 cells, 2000 mAh)
Duration	0.5 hours
Kicking Device	DC motor × 1 (Faulhaber, 24.0 V) Torsion spring × 3
Sensors	Omni-directional camera DC motor encoder × 3 Magnetic Direction sensor

Table 3-2 Risk assessment of the “Musahsi” robot

危険源 (危険状態, 危険事象)		危害のひ どさ	発生 確率	回避 可能性	リスク 評価
押しつぶ し	移動機構に 轆かれる	重い	低い	大	I
切傷	ロボットの 鋭利部と接触	重い	高い	大	II
こすれ, すりむき	ロボットの 鋭利部と接触	重い	高い	大	II
衝撃	ロボットが直接 身体に接触	重い	高い	大	II

## 第4章

ソフトウェアの知能化の例 1 :

照明環境適応型色認識

アルゴリズムの開発

## 4.1 目的

本章では，全方位移動型サッカーロボットのソフトウェアの知能化を目的とする．前章で説明した，ロボカッサッカー中型リーグは，完全自律分散型の移動ロボットの協調行動問題の良いテストベッドであり，基本的な課題が多く盛り込まれている．この中で視覚情報処理は，課題を達成する為の重要な位置を占めており，モータレベルの下位の制御から，ナビゲーション・パスプランニングの上位の行動レベルまで多種多様に関わっている．その為，ロボットが物体を認識する為の色認識アルゴリズムの開発は，重要な課題の一つである．本研究では，視覚情報処理に対してニューラルネットワークを用いた知能化を行うことでより高度な自律化を達成することを目標とする．

## 4.2 脳型計算処理を用いた照明環境適応型

### 色認識アルゴリズムの開発

#### 4.2.1 研究背景

多くの人々にとって見えるということは日常的なことである。多くの人々にとって、見ることなしでは、対象物を知り、その位置関係・動きなどを判断することは容易ではない。視覚の中で、色は役に立つ重要な要素の一つである。日常生活の中で魚の眼の色で、鮮度を見分けたり、サンマの口先の色で油のりを見分けたり、野菜の葉先の色で鮮度を見分ける際にも色情報は役に立っている。また、色を認識することによって路線図や様々なグラフなどに対しても安易に理解することが可能である。色は、視覚刺激のいろいろな特徴の中でも目立つ特徴量である。Gegenfurtner と Rieger[4-8]は、ヒトを被験者として遅延見本あわせ課題をテストすると、色の付いた写真の識別と記憶は、モノクロ写真の識別や記憶よりも良い成績を残すことが報告されている。このように色を認識することは生態系で生活する上で重要な要素である。また、光は波の性質を持ち、その周波数は連続的である。色認識は、ある周波数範囲の光の周波数を動物が見分けるときに脳が行う巧妙な処理の一つである。色が物理世界に存在するということではなく、波長の違いを脳が色としてみているのである。つまり、色を見るのは脳であるということである。一般にヒトは外界を自らの感覚器官を通して受容する。その視覚の中で視覚情報の空間的配置や位置の変化や形は外界に客観的に存在するので、自らが外界へ働きかけ、結果をフィードバックすることで外界と矛盾しない視覚像を脳内に作ることが出来る。

色情報の生物学的な抽出過程は、網膜を通る光を神経細胞が扱う電気信号に変化する視細胞が存在し、それを基に抽出を行う。視細胞には形態の異なる、杆体と錐体が存在する。杆体は、光感受性が非常に高いが光に感受性を持つ物質は一種類しかない。一方、ヒトの網膜の錐体は普通3種類存在し、短波長（青）視物質、中波長（緑）視物質、長波長（赤）視物質のうちいずれかを持っている。この3種類の錐体の活動の違いから光の波長を認識し、色を区別することが出来る[4-9,10]。

以上のように神経活動を行うことによって人間はフレキシブルな視覚システムを備えている。この神経細胞を真似たシステムをニューラルネットワークを用いて行う事によって、より柔軟な対応の出来る Vision システムを開発できるのではないかと予想できる。

ロボティクスの分野においてカメラによる外界の情報はセンサ情報として重要な役割を果たしている。例えば、ロボカップ中型リーグにおいてロボットは、完全に自律型で移動し、サッカーを行う為に様々なタスクを行う。そのタスクの中で重要な課題のひとつは、サッカーを行う際の目標物（ボール、フィールド、白線）を色で認識する事である。また、ロボカップ中型リーグでは、年々色認識に関する条件が高度となり、現在では天気の変化等で時々刻々と照明環境が変化する環境での色認識が求められている。

色認識の中で色変化を取り扱った研究は、大きく分けて2つのアプローチがある。一つ目のアプローチは、対象物の体表面の反射モデルを定義する事でそのパラメータを画像から推定する手法である[4-2,3,4]。このアプローチは、反射光を鏡面反射と拡散反射の2つの反射成分の輪として表す2色性反射モデルが一般的に用いられる[4-5,6]。しかし、鏡面反射を正確に推定するには、光源から発生する鏡面の反射光を性格にカメラ等で推測する必要があり、そのためリファレンスをとるカメラが必要等、複数台のカメラが必要であり、視点や光源環境が変化したときに新たにリファレンスを取らなければならない等の問題点が生じる。つまり、カメラの配置に制限のあるロボットに搭載することや計算コストを考えるとリアルタイム処理の必要な移動ロボットへの適応は困難である。2つ目のアプローチはカメラを用いた直接観測により、対象物周辺の光源を推定する手法である。出村らによると、UrVr空間を用いて対象物の色空間を構築し、空間内の各色の分布の平均・共分散からマハラノビス距離を求めることで、照明の変化にロバストな物体認識の手法を提案している[4-7]。このように、ロボットが様々な照明環境かでもロバストに自動調節できるビジョンシステムの開発が望まれている。

## 4.2.2 目的

本研究では、様々な環境を想定して、それらの環境に適した色認識を行う事が出来るビジョンアルゴリズムをニューラルネットワークなどの学習アルゴリズムを用いる事で構築し、非線形的な照明環境に対してロバストに対応できるアルゴリズムの開発を目指している。

はじめに、様々な色を環境が静的に変化する中で7 or 15 色の色を認識できるか USB カメラを用いて実験した。実験には、学習アルゴリズムとして、Multi-Layer Perseptron (MLP)とモジュール型自己組織化ネットワーク(mnSOM)を用いてアルゴリズムを構築した。この実験によって、ニューラルネットワークを用いたアルゴリズムによって、様々な環境の色を判別する事が可能かを探る。

次にこのシステムを踏襲し、ロボカップ中型リーグ用プラットフォームに用いてサッカーという実環境下での問題に対してロバストに色の閾値を変化させる事が出来るシステムの開発を目標とする。実験には、学習アルゴリズムとして、自己組織化マップ(SOM)、ニューラルガス (NG)、モジュール型自己組織化マップ(mnSOM)を用いた。実験は、静的な環境での実験とロボットが動いている動的な環境での実験を行った。

## 4.2.3 USB カメラを用いた mnSOM を用いた静的環境における照明環境適応型色認識アルゴリズムの検討

### 4.2.3.1 実験環境

実験環境を Fig.4-1 に示す. Fig.4-1 に示すように背景のグリーンの中にある円の色が何であるかを識別する. この際に, 識別に使うパラメータは識別する色の中心の RGB 色相系の値( $R_c, G_c, B_c$ )とバックグラウンドの情報 ( $R_s, G_s, B_s$ ) を入力変数として使用する. なお, 出力はその色の場合を 1 で表し, その色でない場合を -1 で示す. よって, 求める出力値は,  $O = (o_{red}, o_{blue}, o_{yellow}, \dots, o_{black})$  [ $O = -1, 1$ ] というような形になる. この時, 推定したい色のベクトルのみが 1 を示し, それ以外は, -1 という出力を示すものとする. なお, 色は 7 色の場合を {オレンジ, 青, 黄色, 黒, 白, マゼンタ, シアン} とする. 15 色の場合は, {オレンジ, 青, 黄色, 黒, 白, マゼンタ, シアン, 赤, 灰色, ピンク, パールオレンジ, 茶色, 藍色, 黄緑, 緑} である. 実際に使用した画像は, Fig.4-2 に示す. 画像は, USB カメラを用いて撮影した.

色は, 一般的に市販されている絵の具などと同じ色を基準とする (Table 4-1 を参照). Table 4-1 に示す色のパラメータを用いて, レーザープリンタによって印刷したものを実験に使用した.

Table 4-1 に示す上から 7 色 (白い部分) もしくは 15 色 (すべての色) を用いて実験を行う. 撮影した状況 (30 クラス) と照度を Table 4-3 に示す. Table 4-3 における黒文字は訓練データ (27 クラス), 青文字は, テストデータ (3 クラス) とする.

また, 実際に撮影した画像の例を Fig.4-3 に示す. Fig. 4-3 に示すように様々な照明環境下で撮影した場合同じ色でも, 照明の環境によって色の見え方が全く違う. この中から訓練データとテストデータを選択しシステムに組み込むとする. システムについては次節で説明する.

### 4.2.3.2 実験手順

#### (1) MLP を利用したビジョンアルゴリズム実験

MLP を用いたアルゴリズムは，入力層が 6，中間層が 10，出力層が 15 または 7（認識したい色の種類数によって変化）とする(Fig. 4-4)．このネットワークに対して，環境(Table 4-2 参照)を変化させたときにどれくらい認識できるかを実験する．実験には，教師データとし 184 or 405 サンプル（27 環境×色の種類数）を入力して学習を行った．学習後，教師データと未学習データでどれほどの認識率があるかを実験する．学習は，2章で示したバックプロパゲーションの学習アルゴリズムに準ずる．学習率  $\eta$  は 0.6，慣性項は，0.0 と設定した．

#### (2) mnSOM を利用したビジョンアルゴリズム実験

mnSOM は，2章で示した MLP-mnSOM モジュールを応用してシステムを構成している．モジュール内は，SOM で用いられる参照ベクトルと MLP を併せ持っているシステムとなっている．ベストマッチングモジュールの選択には，SOM と同様に参照ベクトルとのユークリッド距離を比較し最小のものをベストマッチングモジュールとする．次に，ベストマッチングモジュールと選択された内部に存在する MLP を用いて色を推定する．この MLP は，入力層が 3（画像の中心の RGB 値），中間層を 10，出力層を 15 or 7(認識したい色の種類数によって変化)とする(Fig. 4-5)．このネットワークに対して，環境(Table 4-2 参照)を変化させたときにどれくらい認識できるかを実験する．実験には，教師データとし 184 or 405 サンプル（27 環境×色の種類数）を入力して，学習を行った．学習後，教師データと未学習データでどれほどの認識率があるかを実験する．学習は，2章で示したバックプロパゲーションの学習アルゴリズムと SOM の参照ベクトルの更新アルゴリズムに準ずる．なお，学習に使用した各パラメータを Table 4-3 に示す．

### 4.2.3.3 実験方法

実験は 30[cm]の距離から撮影した USB カメラの画像をもとに色認識を行う。教示データは USB カメラからの画像を元に様々な環境下 (27 個のクラス) を用いて実験する。Fig. 4-2 で示した入力データ (Rc, Gc, Bc, Rs, Gs, Bs) を元に学習データより学習を行い、未学習データ (3 個のクラス) を用いて実験を行う。

#### 【実験 1】

使用するデータ： 7 色 (データ数) 27 クラス (教示データ)

7 色 (データ数) 3 クラス (教示データ)

{クラス 3, 18, 27}

実験に使用するニューラルネットワーク： バックプロパゲーション,  
MLP-mnSOM

学習回数： 10000, 20000, 30000, 40000, 50000, 100000, 200000 回の認識率を表示

#### 【実験 2】

使用するデータ： 15 色 (データ数) 27 クラス (訓練データ)

15 色 (データ数) 3 クラス (訓練データ)

{クラス 3, 18, 27}

実験に使用するニューラルネットワーク： バックプロパゲーション  
MLP-mnSOM

学習回数： 10000, 20000, 30000, 40000, 50000, 100000, 200000 回の認識率を表示

#### 4.2.3.4 実験結果

##### (1) 実験1の結果

実験1の結果を以下のTable 4-4,5,6,7に示す。Table 4-4,5,6,7に示すように教示データを学習後にテストしてみたところ、バックプロパゲーション単体もMLP-mnSOM アルゴリズムの両方においてかなり高い確率で認識できることがわかる。未学習データに関しては、サンプル数が少ない事を考慮しなければならないが、MLP-mnSOM を用いると正答率が上昇することが分かる。

Figs. 4-6,7に7色のデータを用いて行った実験のBP学習の結果と識別不可能だった画像を表示する。このとき、正しく認識したデータに関しては赤丸で表示しており、誤認識したものは黒丸で表示するようにしてある。一列が1つの照明環境（左からオレンジ、青、黄色、黒、白、マゼンタ、シアン）となっている。図を見て分かるように誤認識した画像は、暗くて見えないなどの特徴がすべてのデータで現れている訳ではない。照度の暗い画像（上から12番目のクラスは真っ暗）に対しても認識できていることが分かる。これはバックグラウンドの閾値と中心の閾値両方が若干ではあるが異なっているため、偶然認識できた可能性がある。

Fig.4-8では今回使用したMLP-mnSOMの環境を特徴マップとして表現したものを示す。特徴マップの格子内にラベルが貼付けられている文字は、場所と照度を示している。照度を考慮すると右下部分には暗い照度のものが集まり、対称的に左上には明るい環境がマッピングされている。また、同じ場所を取った画像は近い位置に分布しており、環境が似ていることが特徴マップより分かる。なお、特徴マップの色は近傍のモジュールとの距離を表しており、青→緑→赤となるにつれて近くから遠くになっていることを表現している。

## (2) 実験 2 の結果

実験 2 結果を以下の Table 4-8,9,10,11 に示す. Table 4-8,9,10,11 に示すように教師データを学習後に教師データのテストを行うとバックプロパゲーション(BP)と MLP-mnSOM 両方のアルゴリズムとも実験 1 に比べてかなり正答率が低くなったことが伺える. 未学習データに関してもサンプル数に対して正答率が低く 50[%]程度の正答率である. BP システム・MLP-mnSOM のシステムの両方とも同じ認識性能であり, 実用的ではない正答率である.

以下に, 実験 1 と同様に BP の訓練データをテストした結果を Fig.4-4 に示す. 一列が 1 つの環境を示し, 正確に出力されたものを赤丸で, 誤認識したものを黒丸で示す. Fig.4-9 をみて分かるように 10 列目 (ピンク), 11 列目の色 (パールオレンジ) の認識率が極端に低い, これは, Table 4-1 に示すように色相に関してはほとんど差がなく, 彩度は光の受け方によってほとんど認識できないことを意味している.

また, 同様に BP の未学習データを実験した結果を Fig. 4-10 に示す. 10 列目 (ピンク) と 11 列目の色 (パールオレンジ) の認識率が低いことが分かる. また, 一番上のクラス (環境 3, Table 4-2 を参照) の認識率が低いこともわかる.

次に, MLP-mnSOM を用いて行った実験に作成された特徴マップを Fig. 4-11 に示す. この図を見て分かるように実験 1 と同様に環境を認識したマップが生成されている. しかし, 今回の実験では環境を認識する中で与えた教示データ以外のモジュールと未学習データとして与えたデータが上手く手こうしなかったと思われる. よって, 認識率が低下したと推測できる.

#### 4.2.3.5 考察

今回の実験を通じて、ニューラルネットワークを用いたビジョンアルゴリズムの可能性を示した。今回の実験では照度の低いデータや高いデータをテストデータとして導入したため、難しい課題ではあったが7色に関してはMLP-mnSOMを用いて環境認識を行えば80%以上認識可能であるという性能が示された。今回使用した色(7色と背景色)は最低限Robocup中型リーグで必要となる色である。よって、必要な色の閾値を出力することができれば試合環境下でも十分に使える可能性がある。

しかし、今回の実験では出力に関しては色の識別を1.0 or -1.0で出力した。ここで考慮しなければならないのは、このシステムで果たして本当にロボットに組み込むことが出来るかどうかである。実際にロボットに組み込むには、色認識を閾値によって設定し範囲を設けて、ロボットに出力するシステムが必要である。よって、今回のモジュールを改良して閾値などを出力するシステムへの改良が必要となる。また、ロボットに組み込むには様々なことを考慮しなければならない。そこで、ロボティクスの面からこのシステムを使用するために必要なことを考察する。

今回使用したシステムは、大きなシステムとなっているため計算時間がかかる可能性がある。ロボティクスの面において、画像処理はリアルタイム性が重要な課題となる。そこで本研究で用いたMLP-mnSOMの処理能力をはかる必要がある。実際に現在全方位移動ロボットMusashiで使用しているカメラの場合、1秒間に30フレームの画像を解像度480x480ピクセルで送信することが可能である。つまり、1つの処理に関して、約33[ms]以内での色認識処理ができればロボットに適したビジョンシステムが必要となる。このようなことを考慮した場合、ロボットが動いている状況下でのオンラインでの学習は、現在のシステムを考えると非常に難しい。そこであらかじめ学習させたデータを用いて環境を認識してロボットを行動させることが重要となるであろう。また、現在のシステムを組み込むためには背景色となる色を抽出し、その色によって照度などのキャリブレーションをネットワーク内部で調整する必要がある。今回の場合緑色を背景色としたが、実際に使用する場合にはリファレンスとなる色をロボット内に装着しそのリファレンスをもとに色認識をする方法が良いと考えられる。

次に、実験2について考察する。実験2の結果をみてわかるように15色と識別する色の種類を増やすと、今回のシステムでは実用出来るシステムとは考えられない。これらの誤認識が起こった理由を考察していく。まず、今回の実験で使った色に関して考える。色に関しては似ている色を15色のうち多数用意した。青と藍色、ピンクとパールオレンジとオレンジ、これらの環境に関しては、認識率が低いことがFig. 4-9を見ればわかる。実際のHSV空間において、上に挙げた色の対は、彩度が異なり、色相がかなり近い値を示す。このように彩度のみが変化する色は、識別が困難であると考えられる。また、暗い環境においては色彩を決める範囲が狭いため、判別が困難であるのではないかと考えられる。

また、今回は1エポック（mnSOMにおける1連の学習アルゴリズム）1回に対してモジュール内のMLPネットワークの学習回数は1回として設定した。これによって、学習率は近傍半径が減少すると下がるので時定数 $\tau$ を50とした場合、学習率は早い段階で収束してしまう。よってその後はBMMとなったモジュール以外はほとんど学習しないので、環境の難しい条件ではBMM以外のモジュールが学習不足になっているのではないかと考えた。しかし、以下のTable 4-12,13に示すように1エポックにつきMLPを10回学習させた場合も性能は向上しなかった。よって、他のパラメータの調整やモジュール自体の改良が必要となると考えられる。他に考えられる改良方法としては、モジュール数を増やすことによって多くの環境を推定することが考えられる。これは、システムが大きくなるなどロボットに組み込む場合、多くの問題があるが認識率を向上させるためには有効な手段かもしれない。

今回実験した環境に関しては、Fig.4-3に示すように背景色の色がかなり変化している。リファレンス（入力）となる色情報は多数の点から平均値を用いるとよりロボットに組み込む場合、性能が向上するかもしれない。

以上より多くのパターンの色を認識するシステムを実現するのは現在のシステムでは、非常に難しいと考えられる。よって、これらを実現するためにはモジュールの各部のパラメータ、モジュール内部のニューラルネットワークシステムの再考察、より多くの環境を学習データとして提示するなどを行って再設計する必要がある。そこで、次に全方位移動ロボットMusashiに組み込む為の色認識アルゴリズムを開発した。

## 4.2.4 全方位移動サッカーロボット **Musashi** に搭載した学習を用いた照明環境適応型色認識アルゴリズムの検討

### 4.2.4.1 ロボカップ中型リーグにおける色抽出

ロボカップ中型リーグにおいて、ロボットは、対象物体の認識を色抽出によって行っている。例えば、ボールはオレンジ、フィールドは緑、ラインは白といった具合である。これらの対象物を認識する事で、ロボットは自律的に自己位置の推定やボールをゴールまで運んでシュートするといった一連の動作を行っている。そのため、ロボットが、対象物を色によって認識する事はタスクを遂行する上で重要な要素技術の一つである。照明環境は、窓などが多く存在する環境の場合、天気等の影響を受けるため照明環境に適した色推定手法が必要である。そこで、前節に置いて、ニューラルネットワークを用いた照明環境適応型色認識アルゴリズムを応用して、ロボカップ中型リーグに適したアルゴリズムに改良を行う。

#### 4.2.4.2 従来の色抽出手法

従来のロボットでは、全方位カメラを使用し、色抽出する事によって物体を認識している。IEEE1394 カメラから PC には、YUV 色相系で表現された各色 8 ビットの信号が送られてくる。このとき、画像中からボール  $O$  (オレンジ) を認識するには以下の式が用いられる。

$$O = (V^o \cap H^o) \quad (4.1)$$

$$V^o \in [V_{min}^o, V_{max}^o] \quad (4.2)$$

$$H^o \in [H_{min}^o, H_{max}^o] \quad (4.3)$$

このとき、上付きインデックスは、対象物の色を示す。 $V$  は、YUV 色相系の  $V$  の値であり、 $H$  は、HSV 色相系の色相の値である。この 2 つの閾値によって 2 値化した画像の論理和を取る事によって、ロボットは、抽出された領域をボールとして認識する。同様に、フィールド  $G$  (緑色) とライン  $W$  (白色) は、以下の式によって与えられる。

$$G = U^g \quad (4.4)$$

$$U^g \in [U_{min}^g, U_{max}^g] \quad (4.5)$$

$$G = U^g \quad (4.6)$$

$$Y^w \in [Y_{min}^w, Y_{max}^w] \quad (4.7)$$

$$S^w \in [S_{min}^w, S_{max}^w] \quad (4.8)$$

現在これらの閾値は、試合直前にユーザによって試行錯誤的に数値を決めていた。しかし、照明の環境が時々刻々と変化する中では、固定の閾値では、色認識は非常に困難な状況となっている(Fig. 4-12 参照)[4-11]。

## 4.2.4.2 提案手法

### 4.2.4.2.1 入出力データ

ロボットのカメラ近傍に基準となる4色（緑，赤，青，白）をテンプレートとして配置する (Fig.4-13). この基準4色のテンプレートをカメラで撮影したときに得られる YUV 色相系の値を入力ベクトル  $x$  とする.

$$x = (Y_g, U_g, V_g, Y_r, U_r, V_r, Y_b, U_b, V_b, Y_w, U_w, V_w) \quad (4.9)$$

ここで, YUV の原信号は 8 ビット ([0, 255]) で表現されているが, 式(9)は, -1.0 から 1.0 の範囲で正規化を行っている. 色認識アルゴリズムの出力ベクトルは, ロボカップ中型リーグの物体認識に必要なボールのオレンジ, ラインの白, フィールドの緑を抽出するための各色の閾値である (式 (4.10) ).

$$y = (V_{\max}^o, H_{\max}^o, U_{\max}^g, Y_{\min}^w, S_{\min}^w) \quad (4.10)$$

上付きのインデックスは,  $o$  がオレンジを表しており,  $g$  が緑,  $w$  が白を意味している. 下付き文字は, 上限か下限かを示している. 上記のベクトル,  $x$  と  $y$  の組を学習ベクトルとする.

#### 4.2.4.2.2 自己組織化マップ(SOM)を用いた手法

1つ目の手法として、自己組織化マップを用いた推定手法を提案する。この時に使用するベクトルは、(4.11)式によって与えられる。

$$\theta_i = [x_i \ y_i] \quad (4.11)$$

自己組織化マップは、入力ベクトル  $\mathbf{x}$  によって、照明環境をクラスタリングし、同時にロボットとへの出力を行うベクトル  $\mathbf{y}$  を学習し、近傍ユニットによって近い環境を補間する事を想定して用いた。本稿で用いる自己組織化マップの計算は、以下の4つのプロセスによって示される(2章参照)。Table 4-14 は学習時に使用する関数及びパラメータである。

#### 4.2.4.2.3 ニューラルガス(NG)を用いた手法

ニューラルガスは、ベクトル量子化に用いられるネットワーク構造であり、SOM に非常に似たネットワークである。しかし、SOM は隣接関係が各ユニット同士に存在しているのに対して、NG は、各ユニットが独立的に学習を行うため、自由に移動する事が可能である。そのため、少ないユニットの量で写像関係を発現できる事が予想される。学習のアルゴリズムは、基本的に SOM のアルゴリズムと同様であるが、競合プロセスにおいて、BMU を決定する際に、それ以外の部分にもユニットに対する距離の近さによってランキングを決定する。以下に学習プロセスを示す。なお、アルゴリズムに使用する変数は、Table 4-14 に準ずる。アルゴリズムは2章を参照とする。

#### 4.2.4.2.4 モジュール型自己組織化マップ(mnSOM)を用いた手法

モジュール型自己組織化マップ(mnSOM)は、SOM を拡張したもので、ここで用いる mnSOM はバックプロパゲーションを利用した階層型ニューラルネットワーク(MLP)と SOM を組み合わせた MLP-mnSOM である。前節で示した色認識アルゴリズムを応用して、今回のシステムを作成する。今回提案するシステム

では、mnSOM 内の各モジュールに MLP 及び参照ベクトルが存在する。参照ベクトル  $u$  は、入力ベクトル  $x$  と同じ次元存在する。また、MLP は、入力ベクトル  $x$  を入力として、出力ベクトル  $y$  を教師信号とし、バックプロパゲーションによって学習を行う。

mnSOM の学習アルゴリズムも 4 つのプロセスによって構成されている(2 章参照)。Table 4-15 は学習時に用いる関数を示す。

#### 4.2.4.3 実験方法

実験は、静的な環境(実験 1)と動的な環境(実験 2)を想定して実験を行った。静的な環境では、ロボット本体は、動作させず様々な照明環境を用いて、ロボットが動作する為に必要な色(オレンジ、緑、白)を認識する事が可能であるかを実験した。この時、計 17 種類の環境を学習ベクトルとして与えた(Table 4-16 参照)。その際の照度は、10~18640 [lx]の値を用いた。照明条件としては、蛍光灯のみ、蛍光灯と日光、日光のみ(屋外)、オレンジ色の白熱灯、白色の白熱灯、オレンジ色と白色の白熱灯の両方を用いた。また、未学習データとして、計 8 種類の環境を作成した。その際の照度は、2 ~ 980 [lx]として、蛍光灯のみ、蛍光灯と日光、日光のみと環境自体は、似ているが学習済ベクトルとは別の照度のものを用いた(Table 4-17 参照)。30000 回の学習を各アルゴリズム(SOM, NG, mnSOM)で行い、その際の学習にかかった時間、学習済ベクトルとの勝者となったユニット(モジュール)の距離の最小自乗誤差、ロボットに搭載した際のアルゴリズム全体にかかる CPU 時間、未学習データの認識率を比較検討する。なお、学習は、マップサイズを 10x10 とし、30000 回行った。

動的な環境では、ロボットが動作を行っている最中に照明の環境を変化させたときにロボットが色を認識する為の閾値をどのように変化させているかを観察する。また、各アルゴリズムがどのようにして、閾値を変更しているかを勝者ユニットがどのように遷移しているかを確かめる。使用する照明環境は、蛍光灯、蛍光灯と日光、日光のみという環境を用いた。Fig. 4-14 に示すようにスタート地点から暗い環境、少し暗い環境(日光あり)、明るい環境と照明条件は同じだが照度の異なる環境を移動させる。この際に学習は、400 [lx], 100[lx], 50 [lx], ボーダー付近で学習データを 4 つ作成した。作成した学習データを各アルゴリズムに挿入し、ロボットを矢印の方向に動かしたときの閾値や勝者ユニットの変化を観察した。マップサイズは、6 x 6 とし、3 万回学習させた。



## 【実験 2】

使用するデータ： 学習データ 4 環境 (Table 4-20 参照)

使用した照明条件： 蛍光灯のみ, 蛍光灯と日光, 日光のみ,

実験に使用するニューラルネットワーク：  
自己組織化マップ(SOM), ニューラルガス (NG),  
モジュラー型自己組織化マップ(mnSOM)

各アルゴリズムのマップサイズ： 6 x 6

学習に使用した PC： MacBook (CPU: 2 Ghz Intel Core 2 Duo,  
Memory : 2 GB 667 Mhz DDR2 SDRAM)

比較項目：

学習回数： 30000 回

近傍関数：最大 6.0  
          最小 1.0

時定数： 300

#### 4.2.4.4 実験結果

##### (1) 実験 1 の結果

実験 1 の結果を以下の Table 4-18 に示す。Table 4-18 に示すように、学習にかかる時間は、明らかに SOM と NG が早い。学習済みデータに対するベストマッチングユニットとの平均自乗誤差は、すべてのアルゴリズムで誤差が小さい。ロボットに踏査したときの画像処理の速度を考慮すると SOM と NG が 2 倍ほどのスピードで処理する事が出来るため、SOM と NG が実用的である。最後に未学習データに対する認識率を考えると、ほとんど差はないが多少 NG と mnSoM が優れている。

##### (2) 実験 2 の結果

実験 2 の結果をアルゴリズム毎に示す。SOM を用いたアルゴリズムで勝者ユニット(BMU)の変化を Fig.4-15 に示す。勝者ユニットが環境によって変化していることが読み取ることが出来る。Fig.4-16 に今回の学習で用いられた SOM の特徴マップと勝者ユニットの遷移について示す。学習ベクトルに対して、環境が変化するとユニットを明るい方からくらい方へと移動していることがわかる。また、遷移の途中では境界部分の学習ベクトルの方を經由して暗い環境に移動していることもわかる。次に、Fig.4-17 に色の閾値の遷移について示す。ユニットが変化すると同時に各色の閾値も更新されていることがわかる。暗い環境では閾値を大きく取り、明るい環境では、環境に応じた値へと変更されていることがわかる。次に、NG を用いたアルゴリズムで勝者ユニットの変化を Fig.4-18 に示す。少ない勝者ユニットしか、選択していないことがわかる。Fig. 4-19 に示すように、NG を用いたときの色の閾値の変化を見ると、少ないユニット数の移動しかないが十分に閾値が変化していることがわかる。最後に、fig.4-20 に mnSOM を用いたアルゴリズムで勝者モジュール(BMM)の遷移を示す。mnSOM も同様に少ないモジュール数で近似されていることがわかる。また、Fig.4-21 に今回の学習で用いられた mnSOM の特徴マップと勝者モジュールの遷移を示す。mnSOM では、暗い特徴のモジュール空間は選択されず、明るい空間でモジュールが変化している。しかし、モジュール内のニューラルネットワークが働き閾値は、時々刻々と変更していることが、Fig.4-17 よりわかる。よって、mnSOM の場合、勝者モジュールが同じであっても、そのモジュール内で入力によって微妙に閾値が調整されていることがわかる。

#### 4.2.4.5 考察

今回の実験を通じて、学習を用いた照明環境適応型色認識アルゴリズムの可能性を示した。今回の実験では、静的な環境と動的な環境と2つの実験を行った。静的な環境では、3つのアルゴリズム（SOM, NG, mnSOM）を用いて性能の比較を行った。実験の結果、静的な環境においては、NGが一番理想的なアルゴリズムであると考えられる。まず、学習の速度に関しては、アルゴリズムの性質上、mnSOMがNGに比べて4倍ほど学習に時間がかかる。つまり、将来のオンラインで学習を想定するとNGがベストである。また、勝者ユニットと学習済ベクトルとの最小自乗誤差は、SOMとNGが良い。また、実行モードでのCPU Timeを考えるとSOM, NGを使うのが現実的である。また、未学習のデータに関しては、NG, mnSOMが良いが認識率としては、1つ誤認識がSOMの方が多かっただけであまり差はない。そこで、総合的に考えるとNGが一番性能の良いアルゴリズムと静的環境の実験では言える。また、他の先攻研究と比較検討すると、出村らによって行われた研究成果[4-7]と比べると、認識率はデータが同一ではないため比べる事不可能だが、実行モードでの時間を比べると、出村らの研究では、画像処理の時間に157 [msec]必要であるのに対して、本研究ではNGでは58[msec]と約3倍の速度で計算することが可能である。これは、移動ロボットがリアルタイムで処理をしなければ、状況を把握することが困難であることを考えると非常に良い結果が得られた。また、出村らの研究では背景色がわかっているならば、推定したい色を認識できないという難点がある。つまり、RoboCupのような特定の環境を想定すれば出村らのアルゴリズムは有効であるが、移動ロボット全般を考えると、特定の環境のみでしか動かないアルゴリズムでは不十分である。本研究のアルゴリズムでは、推定したい色の学習データを収集すれば、リファレンスとなる色の変化と出力したい色の変化を様々な形で応用することが可能である。

次に、実験2の動的な環境について考察する。動的な環境を用いて実験を行ったが、すべてのアルゴリズムで環境の変化によって勝者が変化して、閾値が変化していることがわかる。モジュールの変化を考えると、NGとmnSOMが非常に少ないモジュール数で実行されていることがわかる。以上より、モジュール数を少なくすることによって各アルゴリズムは、計算時間が短く済むことを考えるとNG, mnSOMがアルゴリズム上性能が良いと考えられる。しかし、実験1の

結果を踏まえると、実行モードでの CPU 時間を考慮して、NG が最適ではないかと考えられる。ただし、ロボットが細かい照明環境の変化に対して対応しなければならぬのであれば、細かな閾値の変更を可能としている mnSOM が有効かもしれない。

以上より様々な環境において色を認識するシステムを 3 つのアルゴリズムを比較検討することで実験を行った。結果、アルゴリズムの性能、実行時間や学習にかかる時間、静的・動的環境での有用性をふまえて、NG がアルゴリズム上一番性能がよいと考えられる[4-13,14,15]。

### 4.3 まとめ

ソフトウェア的な知能化のアプローチとして、照明環境適応型色認識アルゴリズムの開発を行った。はじめに、USB カメラを用いた静的な環境において、ニューラルネットワークを用いたアルゴリズムを用いることで、7 色もしくは 15 色の色情報を判別できるかを実験した。この実験を通じて、ニューラルネットワークを用いたビジョンアルゴリズムの可能性を示した。実験では、照度の極端に低いデータや高いデータを入れたので難しい課題ではあったが 7 色に関しては MLP-mnSOM を用いて環境認識を行えば 80%以上認識可能であるという性能が示された。そこで、アルゴリズムの出力部分を認識したい色の閾値に変更することによって、様々な移動ロボットへの適用が可能であると考えた。

次に、USB カメラの実験を踏まえ、全方位移動サッカーロボット”Musashi”のシステムに適応するようにアルゴリズムを変更した。このときに 3 つのアルゴリズムを用いて比較検討する事で有用性を確かめた。使用したアルゴリズムは、SOM, NG, mnSOM を用いた。比較項目として、静的な環境（ロボットが静止した状態）での 30000 回の学習にかかる時間、勝者ユニット（モジュール）と学習済データとの最小自乗誤差、実行モードでの CPU 時間、未学習データの認識率等を比較した。比較した結果、NG を用いたアルゴリズムが一番良い性能を示した。また、動的な環境に置いても、環境に適応しながら色認識アルゴリズムが堂際している事を確認した。また、モジュールの変化等を考慮すると NG を用いると少ないユニット数での写像が可能であると示す事が出来た。そこで、NG が最適なアルゴリズムではないかと考えられる。

以上のようにロボットをハードウェア的なアプローチやソフトウェア的なアプローチから知能化して開発する事によって、ロボットは以前使用していたロボットに比べ格段に性能を向上する事が出来た。実際にロボカップ中型リーグに置いて、4年間で国内大会で3度の優勝と1度の準優勝、世界大会ではベスト4（1回）、ベスト6（2回）、ベスト8（1回）に進出する等の成績を収めている。

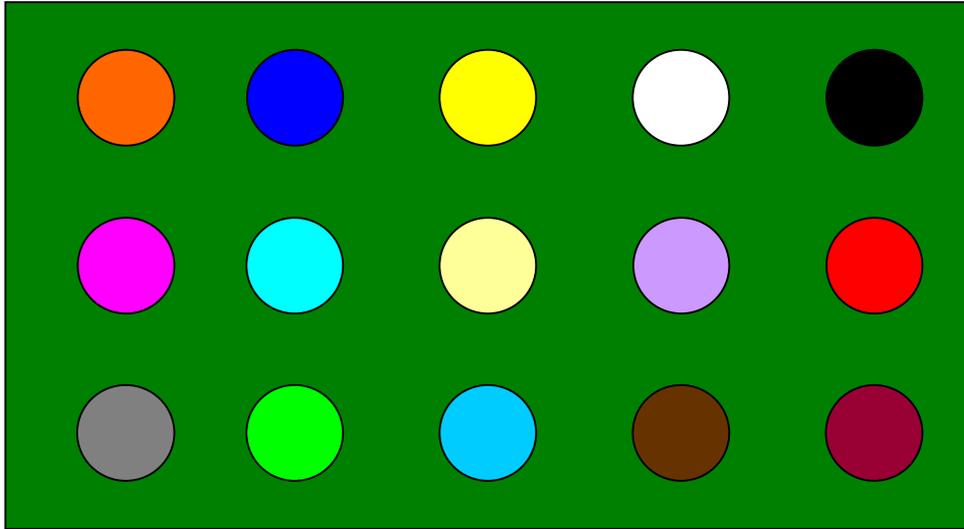


Fig.4-1 Testing data environment

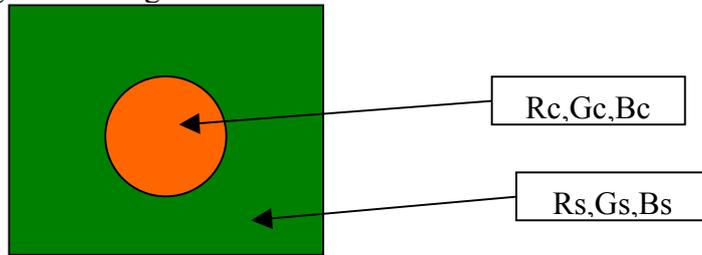


Fig.4-2 Testing data

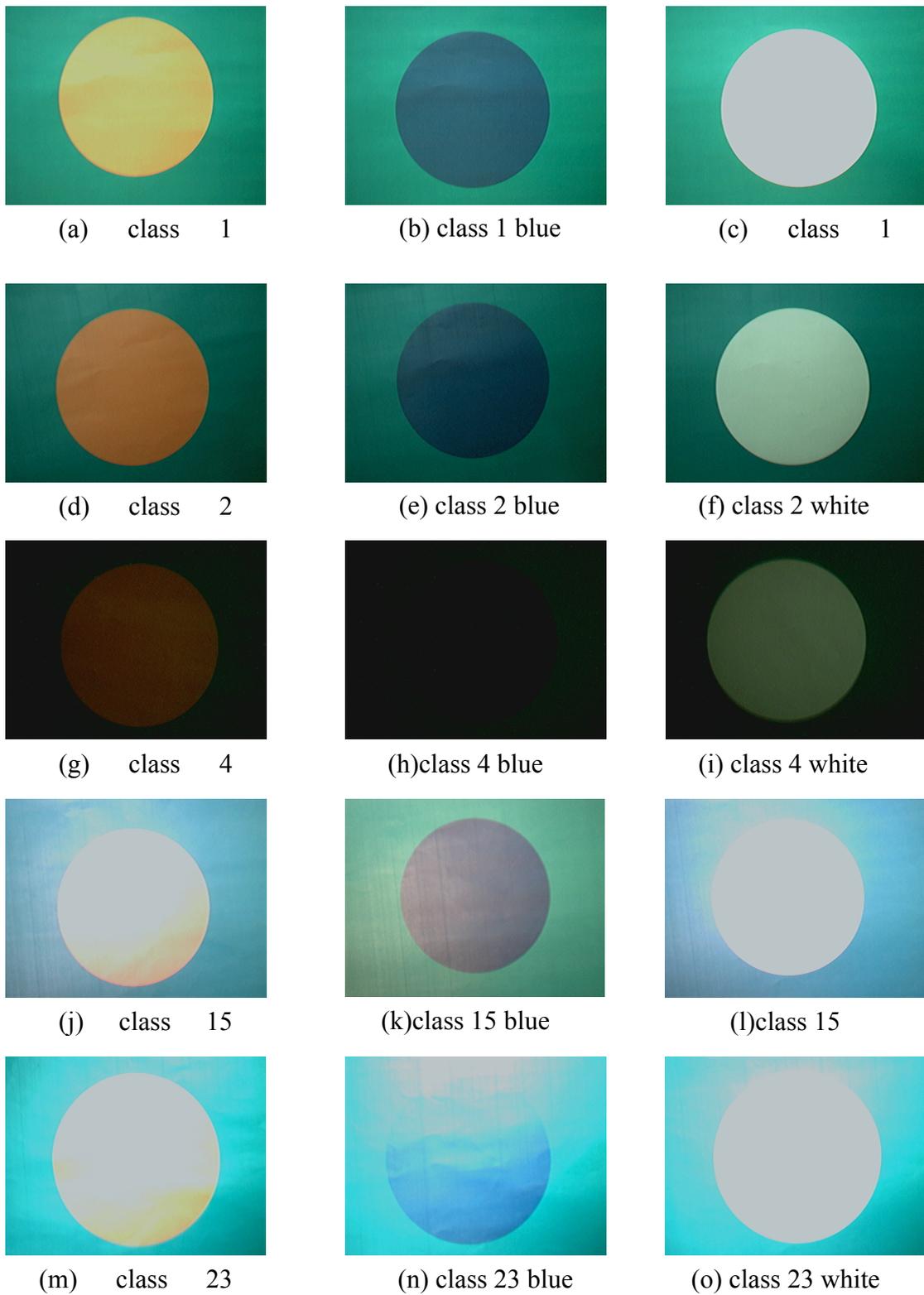


Fig.4-3 Sample of color data

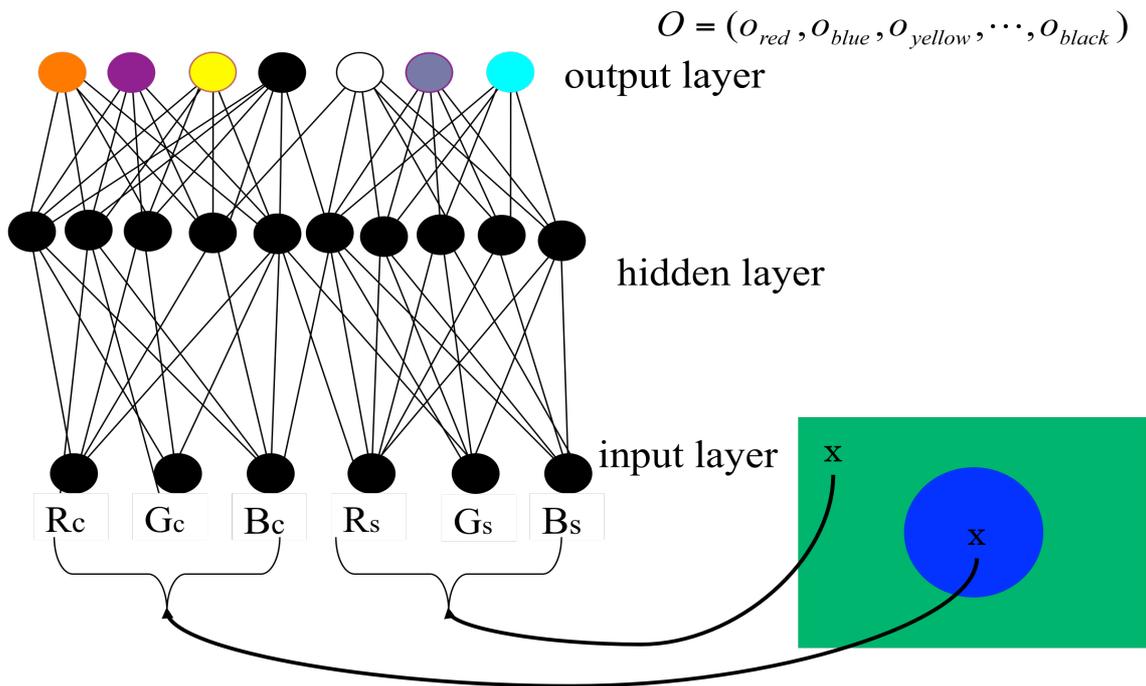


Fig. 4-4 BP vision system

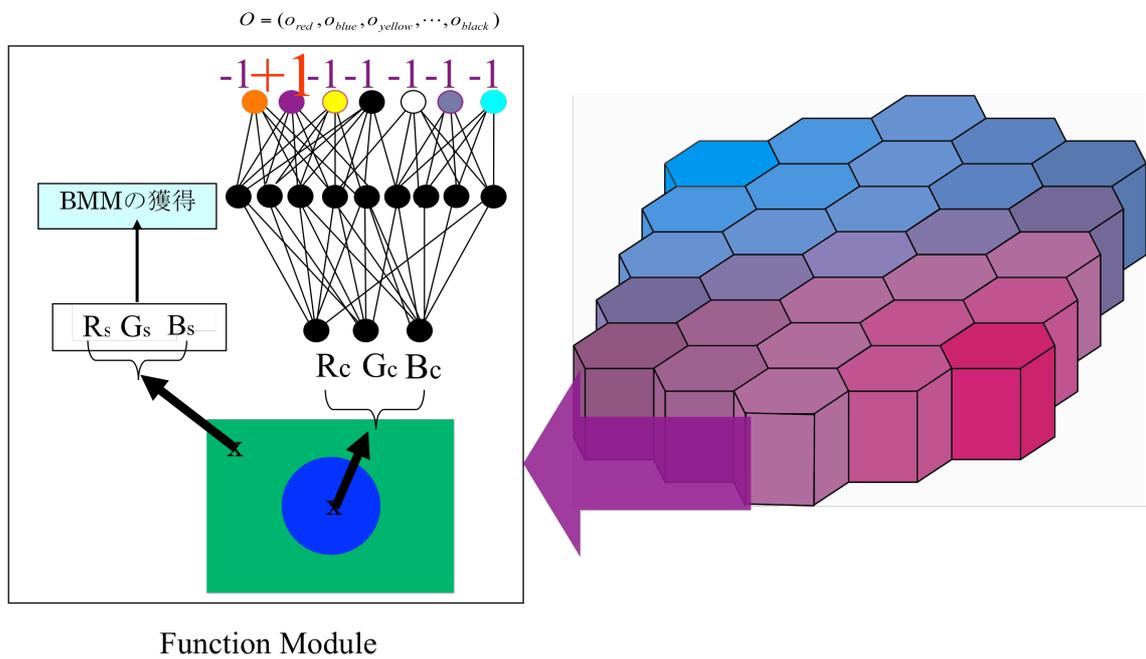
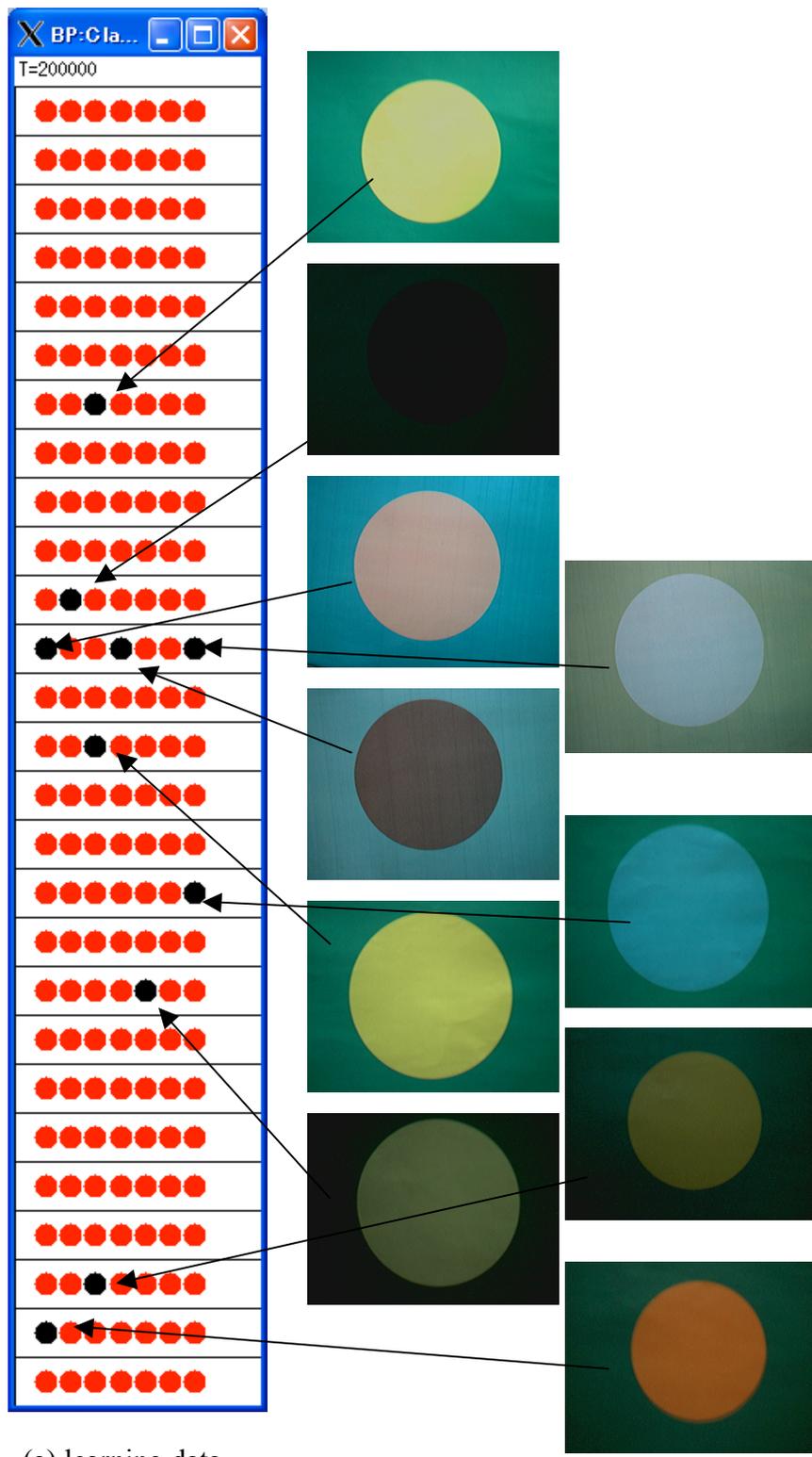


Fig. 4-5 mnSOM's functional module in our algorithm



(a) learning data

Fig.4-6 Result of BP learning (SOMg Canvas) and error color picture

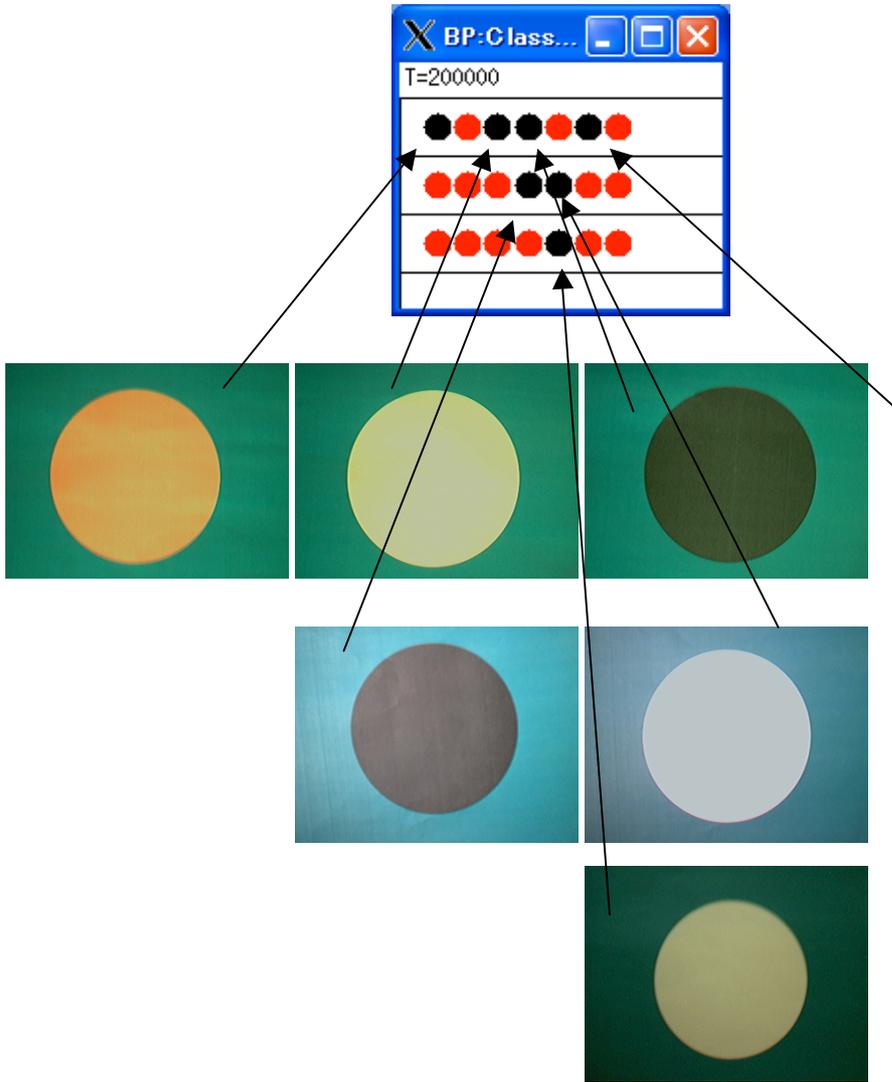


Fig.4-7 Result of BP test data(SOMg Canvas) and error picture

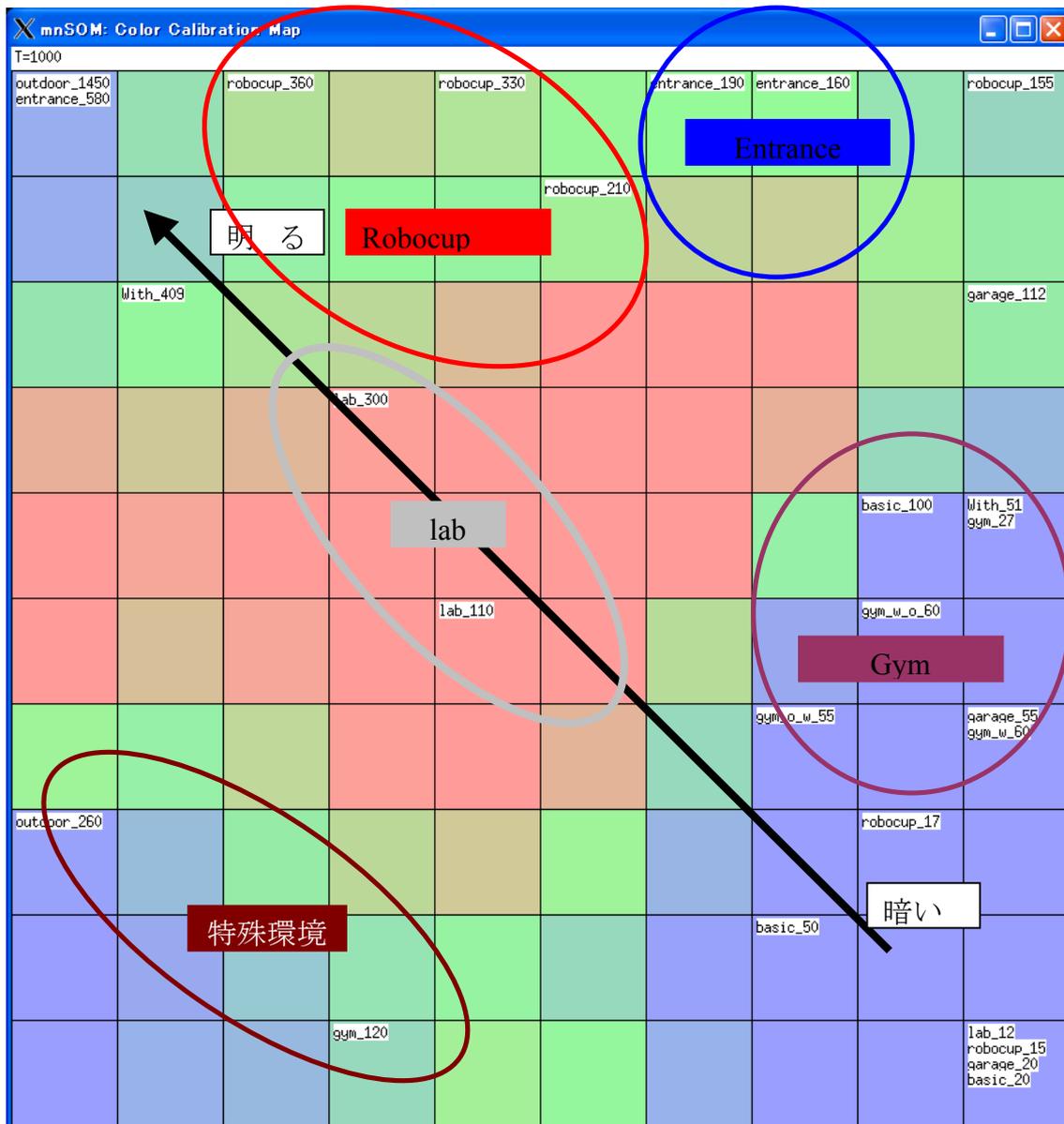


Fig.4-8 Result of mnSOM feature map

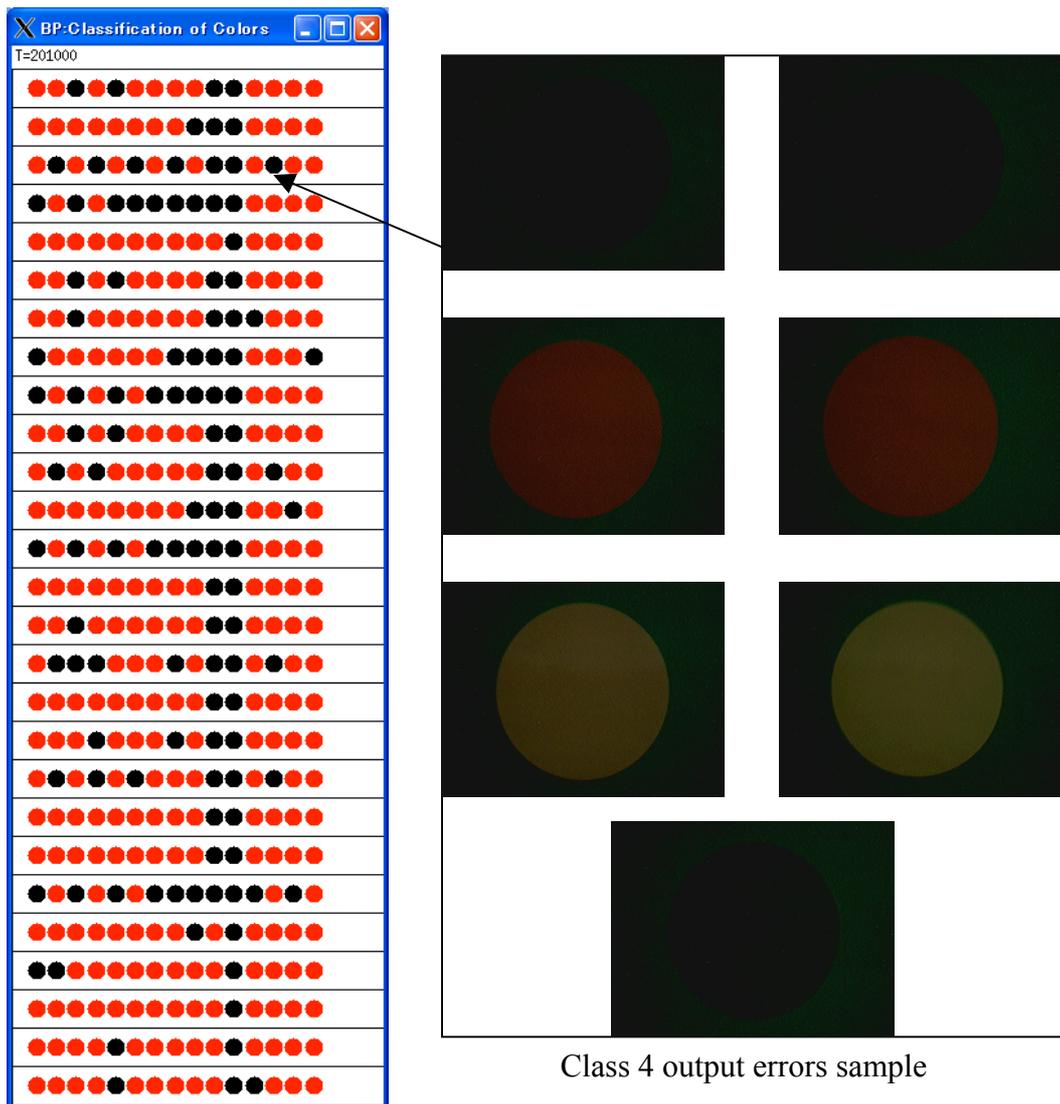


Fig.4-9 Result of BP experiment II learning data error sample

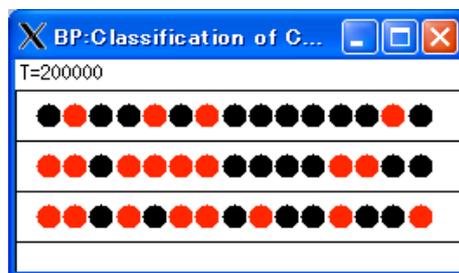


Fig.4-10 Result of BP experiment II test data

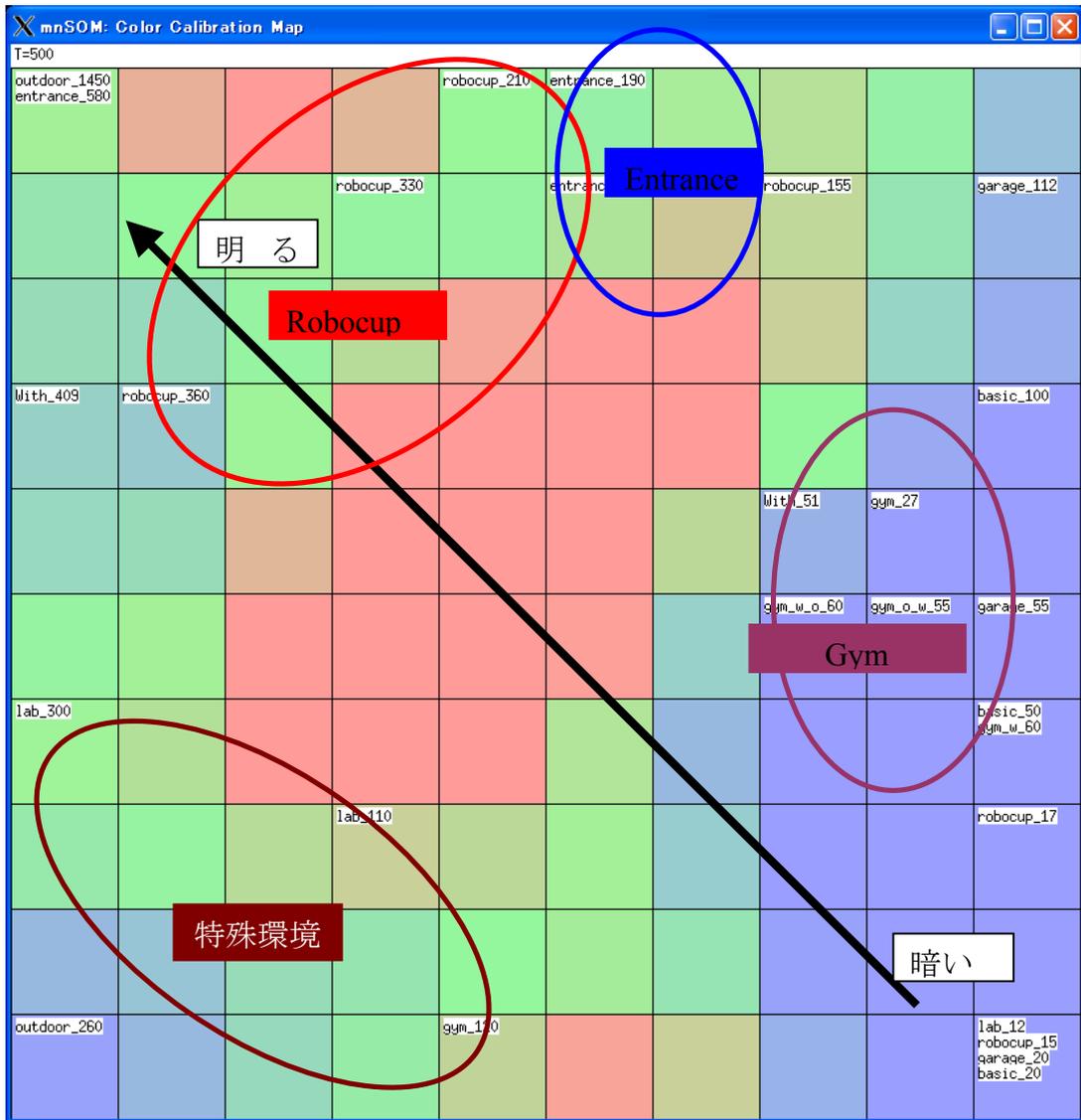


Fig.4-11 Result of mnSOM feature map

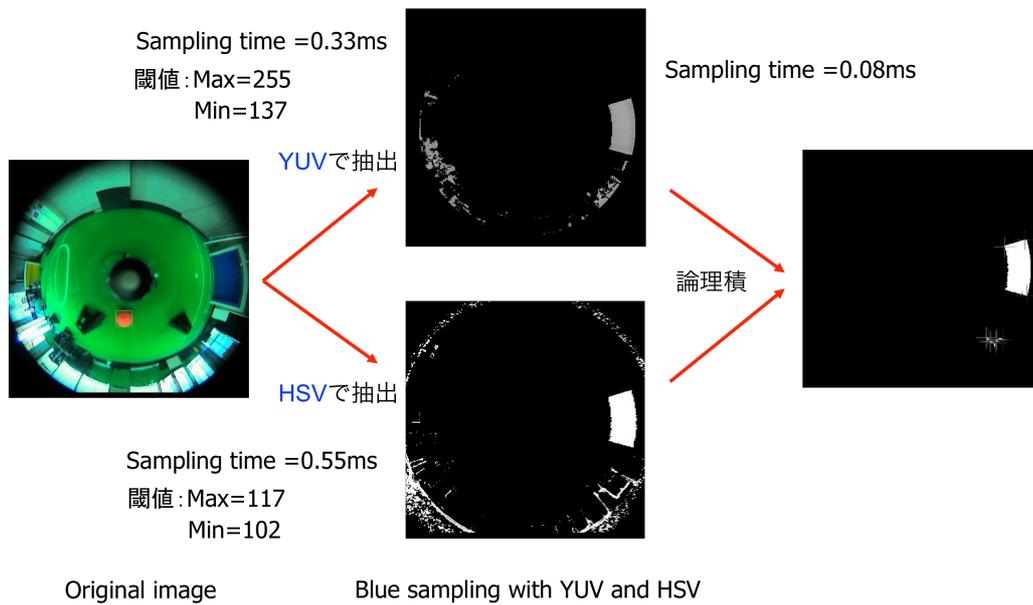


Fig. 4-12 Conventional color extraction algorithm

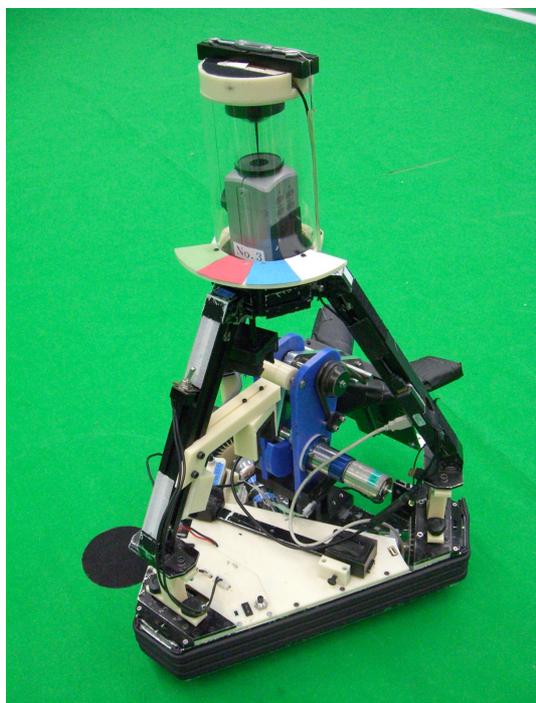


Fig. 4-13 The omni-directional mobile Soccer Robot "Musashi  
Color reference template are put around the omni-directional camera

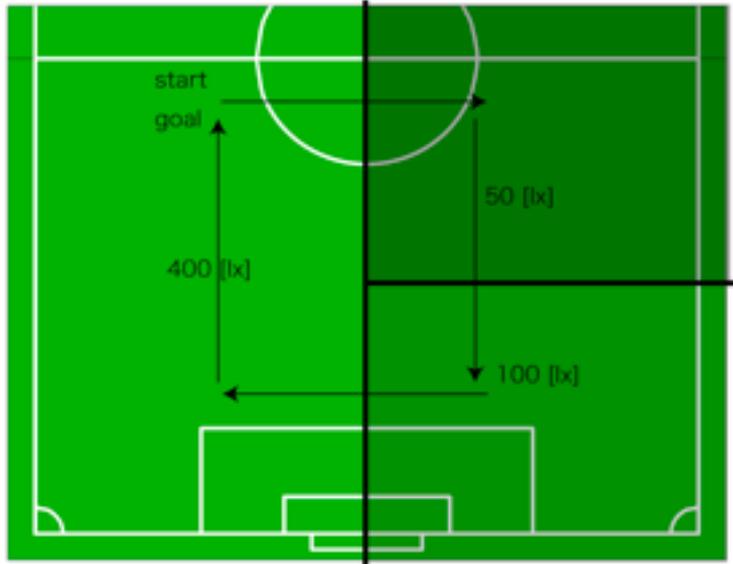


Fig. 4-14 Lighting Condition of the Experiment: Arrow shows the robot movement

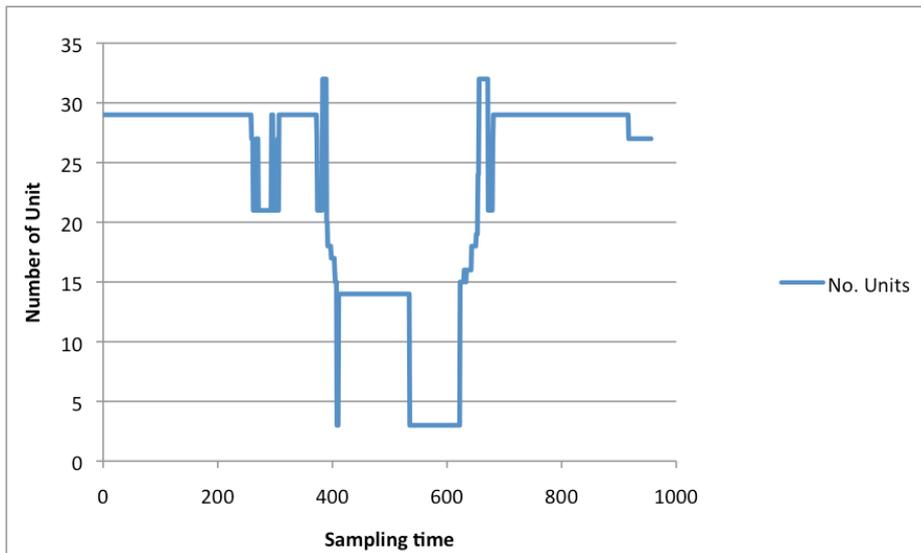


Fig.4-15 Transition the units by SOM algorithm

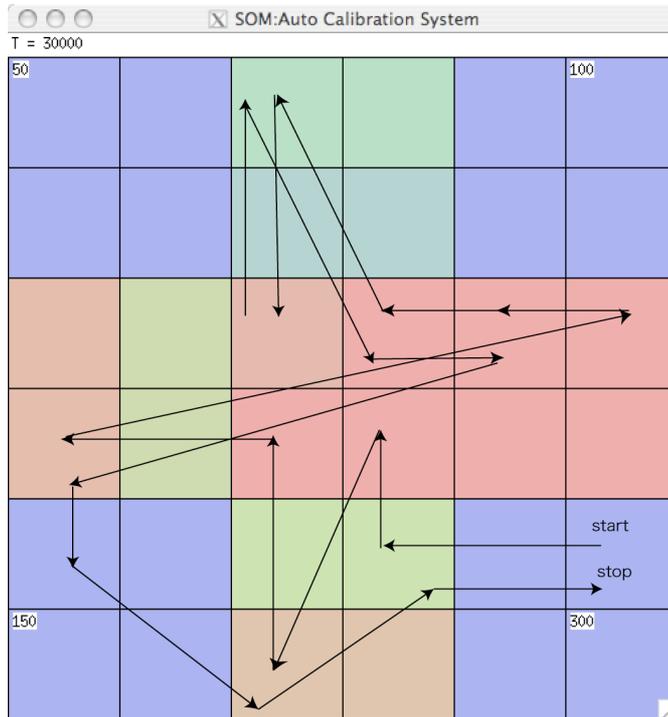


Fig. 4-16 Transition of the unit in feature map by SOM algorithm

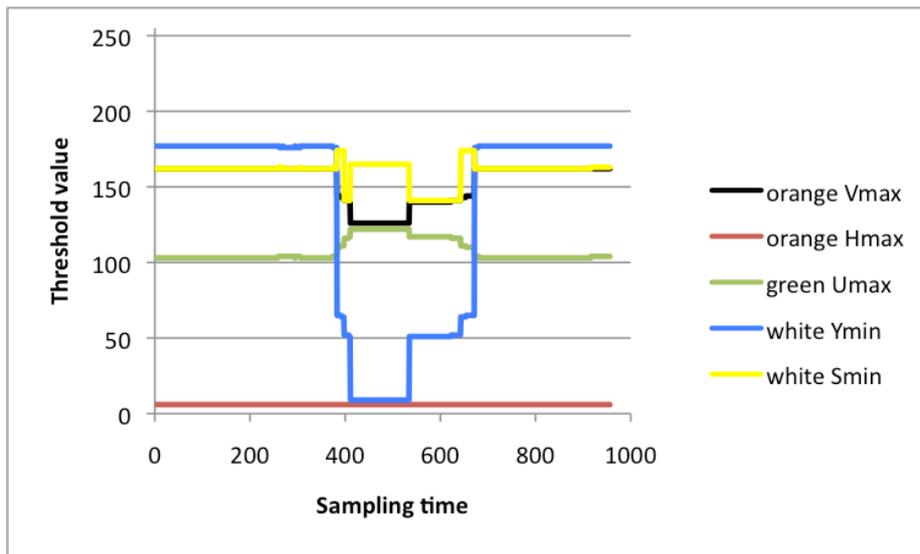


Fig. 4-17 Transition the threshold by SOM algorithm

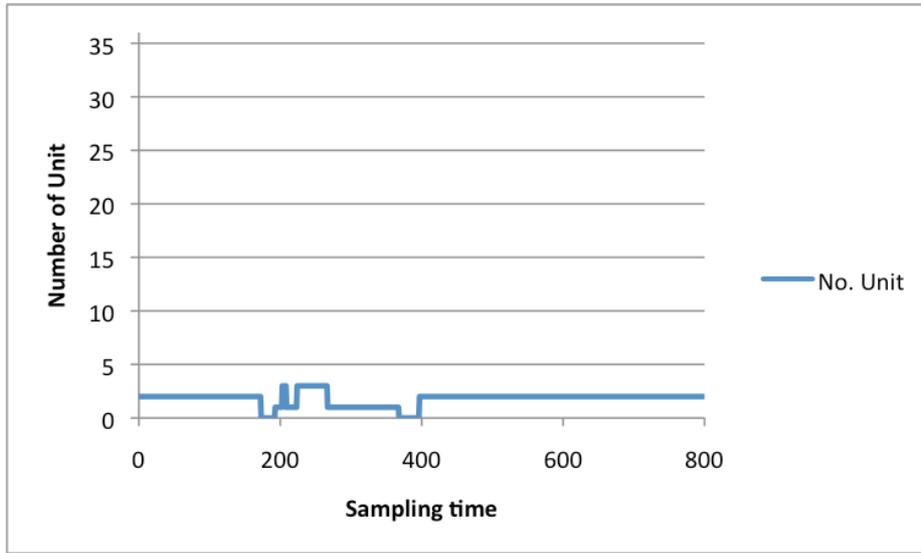


Fig. 4-18 Transition the units by NG algorithm

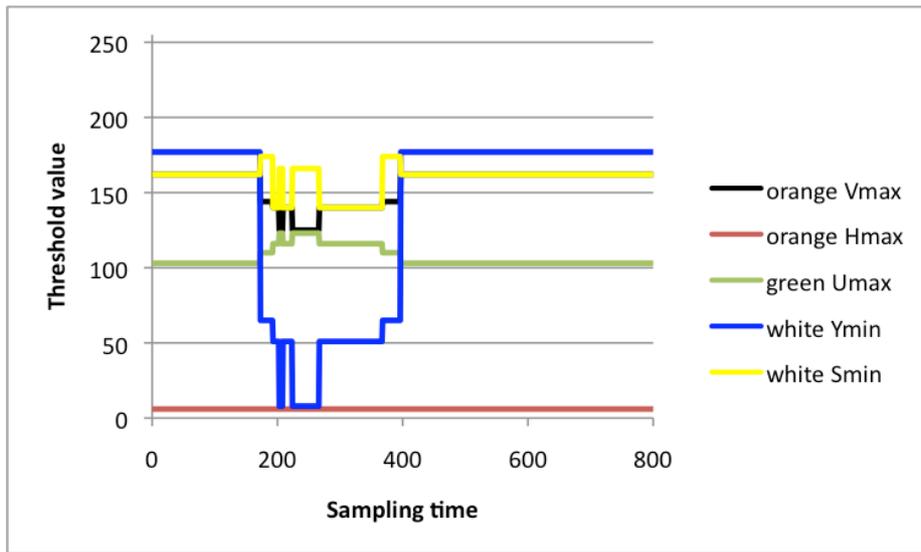


Fig. 4-19 Transition the threshold by NG algorithm

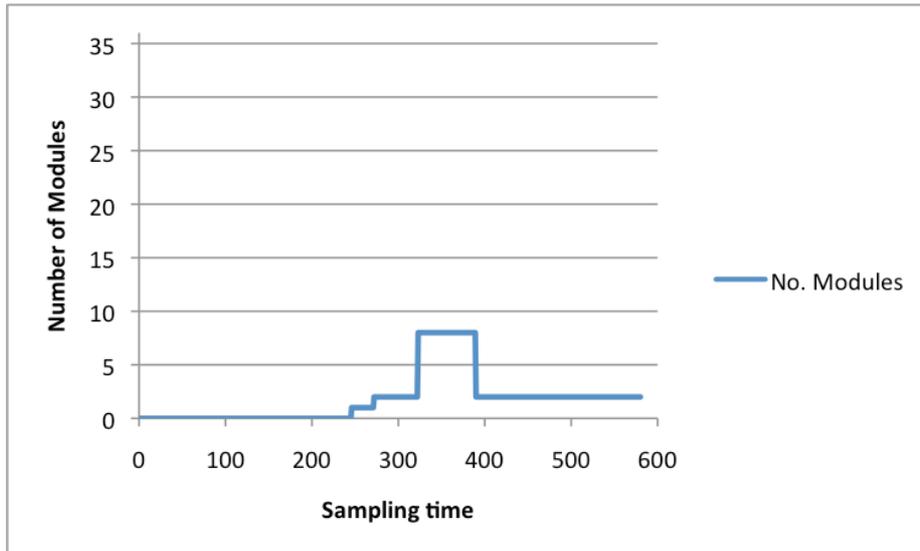


Fig. 4-20 Transition the modules by mnSOM algorithm

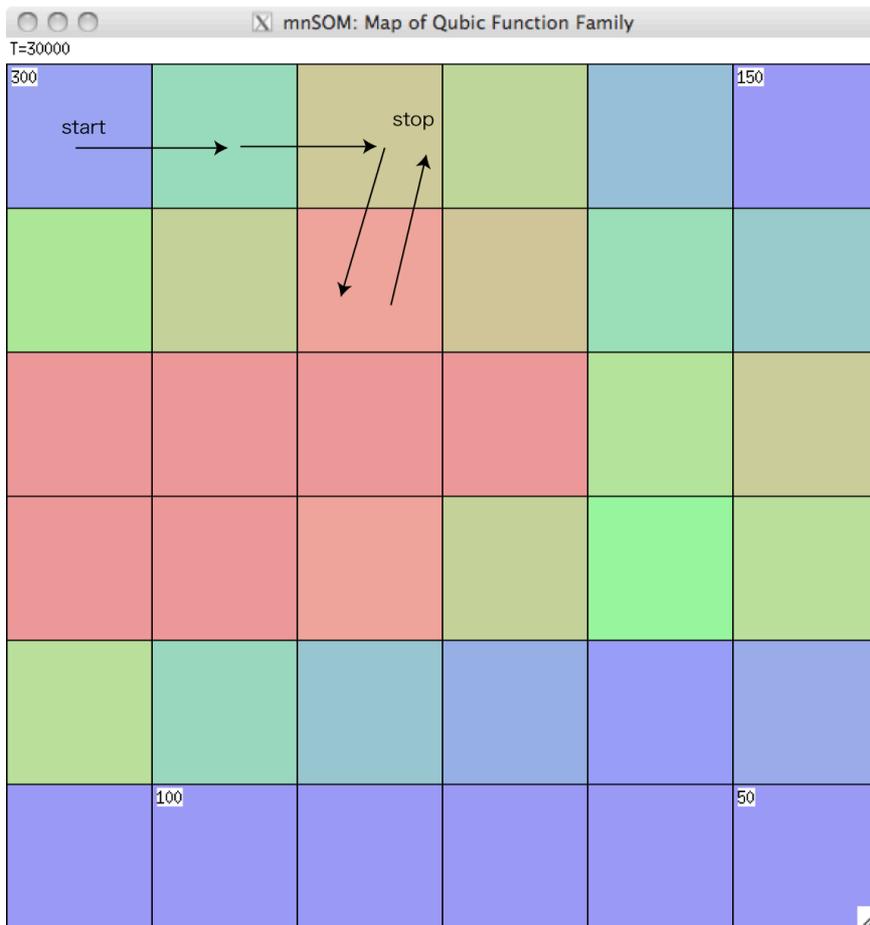


Fig. 4-21 Transition the modules and feature map by mnSOM algorithm

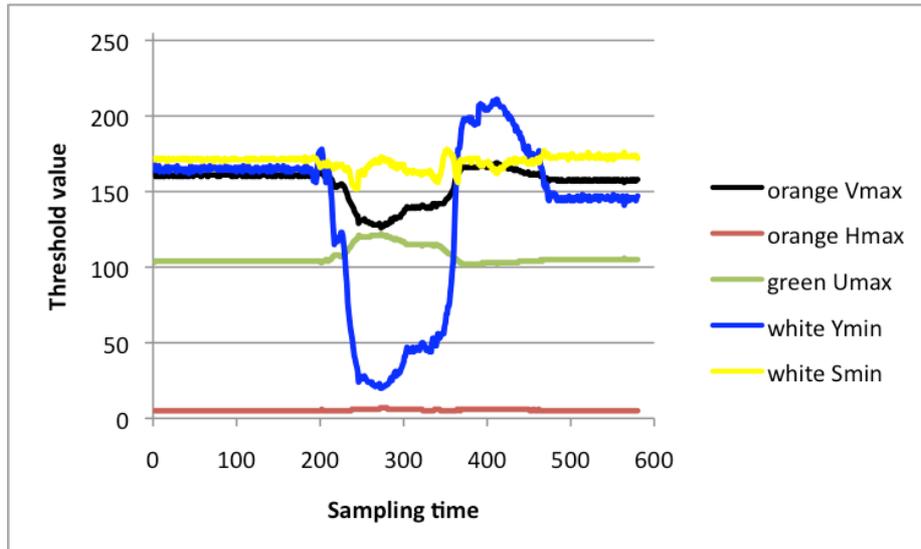


Fig.4-22 Transition the thresholds by mnSOM algorithm

Table 4-1 Test Color

色	H(deg)	S(%)	V(%)
オレンジ	31	94	94
青	240	100	100
黄色	50	100	100
黒	0	0	0
白	0	0	100
マゼンタ	300	100	100
シアン	180	100	100
赤	0	100	100
灰色	0	0	50
ピンク	4	33	97
パールオレンジ	26	39	95
茶色	30	100	60
藍色	207	54	40
黄緑	140	100	100
緑	156	100	71

Table 4-2 Condition of test environment

クラス	撮影場所	暗さ	照度(lx)
1	石井研究室	明るい(外光あり)	300
2	石井研究室	暗い(外光あり)	110
3	石井研究室	中間(外光あり)	180
4	石井研究室	極端に暗い(外光あり)	12
5	COE 部屋	明るい(外光あり)	409
6	COE 部屋	暗い(外光あり)	51
7	ロボカップ部屋	明るい(外光あり)	330
8	ロボカップ部屋	中間(外光あり)	155
9	ロボカップ部屋	極端に暗い(外光あり)	17
10	ロボカップ部屋	明るい(外光なし)	360
11	ロボカップ部屋	中間(外光なし)	210
12	ロボカップ部屋	極端に暗い(外光なし)	15
13	屋外	明るい	260
14	屋外	明るい	670
15	屋外	極端に明るい	1450
16	水中ロボット部屋	暗い(外光あり)	112
17	水中ロボット部屋	暗い(外光あり)	55
18	水中ロボット部屋	極端に暗い(外光あり)	20
19	水中ロボット部屋(basic)	暗い(外光なし)	100
20	水中ロボット部屋(basic)	暗い(外光なし)	50
21	水中ロボット部屋(basic)	極端に暗い(外光なし)	20
22	エントランス	中間(外光あり)	195
23	エントランス	中間(外光あり)	160
24	エントランス	明るい(外光あり)	580
25	体育館	暗い(照明白熱灯[both(orange,white)])	120
26	体育館	暗い(照明白熱灯[white])	60
27	体育館	極端に暗い	27
28	体育館	暗い(照明白熱灯[orange])	55
29	体育館	暗い(照明白熱灯[mix(orange,white)])	55
30	体育館	暗い(照明白熱灯[mix(white,orange)])	60

Table 4-3 Parameters of MLP-mnSOM

モジュール数	10x10
最小近傍半径 $\sigma_{\min}$	2.0
最大近傍半径 $\sigma_{\max}$	15.0
時定数 $\tau$	50.0
安全率 $\beta$	0.9
MLP の入力層のユニット	3
MLP の中間層のユニット	10
MLP の出力層のユニット	7 or 15
学習係数 $\eta$	0.6
慣性項 $\alpha$	0.0

Table 4-4 Result of BP experiment I about learning data

	パターン 数	誤認識数	正解数	正答率	学習回数
訓練データ	189	21	168	88.88889	10000
訓練データ	189	18	171	90.47619	20000
訓練データ	189	17	172	91.00529	30000
訓練データ	189	17	172	91.00529	40000
訓練データ	189	16	173	91.53439	50000
訓練データ	189	13	176	93.12169	100000
訓練データ	189	10	179	94.70899	200000

Table 4-5 Result of MLP-mnSOM experiment I about learning data

	パターン数	誤認識数	正解数	正答率	モジュール数	学習回数
訓練データ	189	23	166	87.83069	100	10000
訓練データ	189	22	167	88.35979	100	20000
訓練データ	189	23	166	87.83069	100	30000
訓練データ	189	22	167	88.35979	100	40000
訓練データ	189	22	167	88.35979	100	50000
訓練データ	189	21	168	88.88889	100	100000
訓練データ	189	17	172	91.00529	100	200000

Table 4-6 Result of BP experiment I about non-learning data

	パターン数	誤認識数	正解数	正答率	学習回数
テストデータ	21	7	14	66.66667	10000
テストデータ	21	8	13	61.90476	20000
テストデータ	21	8	13	61.90476	30000
テストデータ	21	8	13	61.90476	40000
テストデータ	21	7	14	66.66667	50000
テストデータ	21	8	13	61.90476	100000
テストデータ	21	7	14	66.66667	200000

Table 4-7 Result of MLP-mnSOM experiment I about non-learning data

	パターン 数	誤認識数	正解数	正答率	モジュー ル数	学習回数
テストデータ	21	5	16	76.19048	100	10000
テストデータ	21	4	17	80.95238	100	20000
テストデータ	21	5	16	76.19048	100	30000
テストデータ	21	5	16	76.19048	100	40000
テストデータ	21	6	15	71.42857	100	50000
テストデータ	21	7	14	66.66667	100	100000
テストデータ	21	4	17	80.95238	100	200000

Table 4-8 Result of BP experiment II about learning data

	パターン 数	誤認識数	正解数	正答率	学習回数
訓練データ	405	195	210	51.85185	10000
訓練データ	405	169	236	58.2716	20000
訓練データ	405	146	259	63.95062	30000
訓練データ	405	140	265	65.4321	40000
訓練データ	405	134	271	66.91358	50000
訓練データ	405	125	280	69.1358	100000
訓練データ	405	116	289	71.35802	200000

Table 4-9 Result of MLP-mnSOM experiment II about learning data

	パターン 数	誤認識数	正解数	正答率	モジュー ル数	学習回数
訓練データ	405	124	281	69.38272	100	10000
訓練データ	405	106	299	73.82716	100	20000
訓練データ	405	107	298	73.58025	100	30000
訓練データ	405	113	292	72.09877	100	40000
訓練データ	405	110	295	72.83951	100	50000
訓練データ	405	100	305	75.30864	100	100000
訓練データ	405	98	307	75.80247	100	200000

Table 4-10 Result of BP experiment II about non-learning data

	パターン 数	誤認識数	正解数	正答率	学習回数
テストデータ	45	29	16	35.55556	10000
テストデータ	45	29	16	35.55556	20000
テストデータ	45	29	16	35.55556	30000
テストデータ	45	29	16	35.55556	40000
テストデータ	45	25	20	44.44444	50000
テストデータ	45	23	22	48.88889	100000
テストデータ	45	25	20	44.44444	200000

Table 4-11 Result of MLP-mnSOM experiment II about non-learning data

	パターン数	誤認識数	正解数	正答率	モジュール数	学習回数
テストデータ	45	30	15	33.33333	100	10000
テストデータ	45	25	20	44.44444	100	20000
テストデータ	45	26	19	42.22222	100	30000
テストデータ	45	25	20	44.44444	100	40000
テストデータ	45	25	20	44.44444	100	50000
テストデータ	45	26	19	42.22222	100	100000
テストデータ	45	25	20	44.44444	100	200000

Table 4-12 Result of MLP-mnSOM about learning data (10 bp learning/ epoch)

	パターン数	誤認識数	正解数	正答率	モジュール数	mnSOM 学習回数	モジュール 内学習
訓練データ	405	132	273	67.40741	100	10000	100000
訓練データ	405	124	281	69.38272	100	20000	200000
訓練データ	405	116	289	71.35802	100	30000	300000
訓練データ	405	121	284	70.12346	100	40000	400000
訓練データ	405	118	287	70.8642	100	50000	500000
訓練データ	405	118	287	70.8642	100	100000	1000000

Table 4-13 Result of MLP-mnSOM about non learning data ( 10 bp learning/ 1 epoch)

	パターン数	誤認識数	正解数	正答率	モジュール数	mnSOM 学習回数	モジュール 内学習
テストデータ	45	28	17	37.77778	100	10000	100000
テストデータ	45	29	16	35.55556	100	20000	200000
テストデータ	45	31	14	31.11111	100	30000	300000
テストデータ	45	29	16	35.55556	100	40000	400000
テストデータ	45	32	13	28.88889	100	50000	500000
テストデータ	45	33	12	26.66667	100	100000	1000000

Table 4-14 Variables used in Explanation of SOM and NG algorithms

Symbol	Quantity
$\theta$	Learning data
$x$	Input vector
$y$	Output vector
$i$	Index expressing classes ( $i = 1, \dots, I$ )
$w$	Reference vector
$k$	Index expressing unit ( $k = 1, \dots, K$ )
$E$	Distance between input vector and reference vector
$k^*$	Best Matching Unit (BMU)
$\phi$	Neigubor Function
$d(a,b)$	Euclidean distance between a and b
$\psi$	Learning late
$\sigma$	Neighbor radius
$\tau$	Time constant

Table 4-15 Variables used in Explanation of mnSoM algorithm

Symbol	Quantity
$v$	Reference module vector
$x$	Input vector
$y$	Output vector
$j$	Index expressing classes ( $j = 1, \dots, J$ )
$m$	Index of module
$E_m$	Distance between input vector and reference vector
$m^*$	Best Matching Module (BMM)
$\phi$	Neigubor Function
$d(a,b)$	Euclidean distance between a and b
$\psi$	Learning late
$\sigma$	Neighbor radius
$\tau$	Time constant

Table 4-16 Environment of the learning data

Location (light environment)	Illuminance [lx]
RoboCup room (FL)	10
RoboCup room (FL)	53
RoboCup room (FL)	110
RoboCup room (FL)	453
RoboCup room (FL and SUN)	81
RoboCup room (FL and SUN)	90
RoboCup room (FL and SUN)	450
Entrance (FL and SUN)	25
Entrance (FL and SUN)	45
Entrance (FL and SUN)	125
Entrance (FL and SUN)	252
Outdoor (SUN)	228
Outdoor (SUN)	18640
Gym (SUN)	35
Gym (WM and SUN)	113
Gym (OM and SUN)	142
Gym (WM, OM and SUN)	216

SUN: Sunshine, FL: fluorescent light,  
 WM: white mercury lamp, OM: orange mercury lamp

Table 4-17 Environment of the non learning data

Location (light environment)	Illuminance [lx]
RoboCup room (FL)	310
RoboCup room (FL and SUN)	69
RoboCup room (SUN)	2
Entrance (FL and SUN)	337
Entrance (FL and SUN)	168
Entrance (SUN)	30
Outside (SUN)	58
Outside (SUN) [Evening]	980

Table 4-18  
Result of Experiment I

	SOM	NG	mnSOM
Learn speed [sec]	27.9	17.1	81.5
Error	0.0007	0.0000	0.0367
Execution speed [msec]	61.3	58.4	114.9
Recognize Rate[%]	83.3	85.4	85.4

## 第 5 章

### ソフトウェアの知能化例 2 : 自律型水中ロボットの 適応制御システムの開発

## 5.1 緒言

地表上での海洋の面積は、3億6000万km<sup>2</sup>で、陸地の面積1億5000万km<sup>2</sup>と比較すると2.4倍もの大きさである。平均深度は、4750[m]であることを考えると膨大な空間である。日本の近海には、貴重な鉱物資源やエネルギー資源があるとされている。各国の経済発展に伴う銅などの枯渇が深刻化している[5-1]。

海底下の地殻内にアーキアと呼ばれる微生物が存在する事が明らかになってきており、その量は地球上の生物存在量の半分を海底下に占めていると言われている。また、海底には生命の起源や新たな微生物資源の発見も期待されている[5-2]。

このように、海洋開発や科学調査等は、人類に大きな利益をもたらすが、非常にコストがかかるという問題もある。水中とは我々人間にとってどのような環境であろうか。人間は、専用の潜水機材がなければ水中で息をする事は出来ない。また、水深が10[m]増す毎に1気圧増加し、水圧によって減圧症や窒素酔いと呼ばれる窒素中毒に陥る。減圧症は、最悪死に至る場合があり、窒素酔いは、死亡する事はないが、正常な判断が出来なくなる等、人間の精神に異常をきたす。このように、人間に取って水中で活動する事は、常に危険と隣り合わせの状態である。そこで、ロボットに代表されるような無人機械による調査や開発が期待されている。

ロボットにとって水中は、どのような環境なのかを考えると、陸上では電波を通して通信を行う事が出来るが、水中では電波が達してもせいぜい4.0[m]程度と実海域において作業する事は難しい。通常は、超音波等の音響によりロボットとの通信を行って動くが、リアルタイムでロボットを制御できるほど高速な通信はない。同様にカーナビなどで使用される、GPSなどを利用した自己位置同定法は、水中では利用できない。その為、水中ロボットが自律行動を行う為には、非常に高度な知能が必要である。

では、海洋において水中ロボットは、どのような分野で活躍がのぞまれているのであろうか。1950年代に米国海軍が観察専用の無人潜水機の開発を開始し、嫌いの回収を目的として1958年に建造された米国海軍の無人潜水機CURV-I[5-3]は、1996年に米国爆撃機が誤ってスペイン沖水深868[m]に投下した水爆を回収した。その後も、石油生産施設等では設備の検査や水中ダイバーの監

視等で必要不可欠なものとなっている。禁煙では、自律型水中ロボット(AUV: Autonomous Underwater Vehicle)が海底火山の調査を行う等本格的に実用化されている[5-4,5,6,7,8].

以上のように水中で作業を行うには、水中ロボットが必要不可欠な存在となっており、それらの技術も確立してきている。今後は、水中ロボットの高度化や汎用性を挙げる為にもロボットの自律化が大きく望まれている。

## 5.2 代表的な水中ロボット

本節では、過去から現在に至るまでの水中ロボットについていくつか紹介する。水中ロボットには、有人潜水艇を含めるといくつかに分類することができる。分類は、Fig.5-1 に示すように分類することができる[5-9]。では、現在までにどのような水中ロボットが存在するかを以下に紹介する。

〈海外の水中ロボット〉

ALVIN[5-10]は、1964年に米国海軍が開発した有人潜水艇である。所属は、米国海軍であるが、運用は米国のウッズホール研究所(Woods Hole Oceanographic Institution : WHOI)が行っている。潜航回数 3700 回を超え現在も世界で活躍している有人潜水艇のひとつである。全長 7.0[m], 高さ3.7[m], 耐圧容器には、チタニウム合金を使用し、最大水深 4500[m]である。搭乗可能人数は、パイロットも含め3人である。潜航時間は6時間から 10時間である。沈没した TITANIC 号を 1986 年に発見したのは有名な話である。現在もさまざまな研究に運用されている(Fig.5-2).

ABE(Autonomous Benthic Explorer)は、WHOIで1993年に開発されたAUVである。ロボット上部に浮力材が搭載され、下部にLi-ion電池を搭載している。全長3.0[m], 全幅 2.0[m], 高さ 2.5[m], 重量 550[kg]である。最大深度 5000[m]まで潜航可能である。スラストは前進方向に3基, 上下方向に2基, 水平方向に2基搭載している。ABE は、開発されてから 200 回程度のミッションを遂行している。特に 1996年から2006 年にかけて 155 回のミッションを行い、観測距離2500[km], 平

均深度2000[m]で数々の実績を上げている。WHOIでは、現在ABEの後継の水中ロボットを開発中である。このロボットは、本研究で開発しているロボット同様にAUV, ROVとしても使用できるような設定としている。(Fig.5-3)

マサチューセッツ工科大学(MIT : Massachusetts Institute of Technology) Sea Grantでは、1995年にOdyssey IIシリーズ[5-11]を5機製造しておりそのうちの2基を先述の WHOI やサンディエゴ海軍研究所等に分配されている。2基のうちの1機Odyssey IIc は、2000年に改良され、全長 2.2[m], 空中重量約200[kg]である。電源には、Li-polymer バッテリーが採用され 8 時間、44[km]の潜航が可能である。センサには、高画質カメラ、サイドスキャンソナー等が搭載している。運動の制御はエレベータおよびラダーを用いて行う(Fig.5-4).

REMUS(Remotely Environmental Monitoring UnitS)[5-12]は、WHOIで開発されREMUS100, REMUS600, REMUS6000 などがある。REMUS はローコスト型のAUVである。REMUS100 は、全長1.60[m], 重量37[kg], 潜航深度100[m]で ADCP, CTD, サイドスキャンソナーを搭載している(Fig.5-5).

Jason/Medea[5-13]は、Institution's Deep Submergence Laboratoryが開発したROVである。運用は先述した WHOI である。全長 3.4[m], 全幅 2.4[m], 高さ 2.2[m], 重量約 3.7[t]で最大潜航深度 6500[m]である。ロボットは 6 基のスラストにより 制御され、6 自由度のマニピュレータも搭載している。またカメラを 8 基搭載している(Fig.5-6).

カナダの ISE(International Submarine Engineering Ltd.)は、1983 年から自律型水中ロボット ARCS を開発している[5-14]。現在は、AUV, ROV(HYSUB)シリーズとして海底掘削作業サポート ROV, 作業用 ROV などを製品化している。HYSUB5000 は、全長 2.54[m], 全幅 1.52[m], 高さ 1.65[m], 重量約 2.2[t]で最大水深 5000[m]まで潜航可能である。ロボットには、5 自由度と 7 自由度のマニピュレータが搭載されている。また、ISE では、水上無人機などの開発も行っている(Fig.5-7).

韓国の MOERI(Maritime & Ocean Engineering Research Institute), Korea Ocean Research & Development では,2001 年から 6000[m]級の ROV “HEMIRE” の開発を行っており,2006 年に水深 5600[m]に到達している[5-15] (Fig.5-8).

<国内の水中ロボット>

しんかい 6500[5-16]は,海洋研究開発機構で 1990 年に開発された有人潜水艇である。潜航深度は,6500[m]で有人の潜水艇としては世界で最も深く潜航することが可能である。全長 9.5[m],重量 25.8[t]である。しんかい 6500 は,耐压球の直径 2[m]で材料はチタニウム合金である。マニピュレータは,7 自由度のものが 2 基搭載されている。2003 年度にオーバーホールで銀亜鉛電池からリチウムイオン電池に換装された。現在は,年間60回程度の調査,観測潜航を行っている(Fig.5-9).

かいこう[5-17]は,海洋研究開発機構で 1995 年に開発され 1 万メートルまで潜航可能な ROV である。当初の開発目的は,前述のしんかい 6500 の事前調査と救難を行うためであった。しかしながらしんかい 6500 の救助方式が変更となり現在は,支援母船をかいれいとして海溝域の観測を行う潜水機として活躍している。2003 年に室戸岬沖で二次ケーブル破断によりビークルを亡失。現在は,7000[m]級無人潜水機を改造し,ランチャーに搭載してかいこう 7000 として運用中である。ランチャー部の全長 5.2[m],重量 5.3[t]であり最大 11000[m]まで潜航可能である。ランチャーには,サイドスキャンソナーなどが搭載されている。ビークル部は,全長2.8[m],重量2.7[t]である。6自由度のマニピュレータが 1 基搭載されており,カメラが 5 台搭載されている(Fig. 5-10).

r2D4[5-18,19,20]は,東京大学生産技術研究所で 2003 年に開発された AUV である R-ONE(Fig.5-11)ロボットの後継機である。全長4.6[m],重量1.6[t]で最大深度 4000[m]である。電源はリチウムイオン電池で航続距離 60[km]である。実績としては,明神礁のカルデラ観測などを行っている(Fig. 5-12).

Aqua Explorer 2000[5-21]は,KDDI 研究所で開発された海底ケーブルの保守点検を目的とした AUV である。全長 3[m],重量 300[kg]で最大 2000[m]まで潜航可能である。電源は,リチウムイオン電池で連続航続時間は,16時間である。現在は,

ケーブルの点検以外にもハイドロフォンをロボットに搭載し、鯨の追跡等も行っている(Fig.5-13).

Tri-Dog I[5-22,23,24]は、1999年に東京大学生産技術研究所で開発され、実用化を目指したAUVである。全長 1.85[m]、重量 180[kg]、最大水深 100[m]である。Tri-Dog I は、岩手県釜石港において世界初のケーソンの自動観測に成功している[28]。また現在では、鹿児島県錦江湾の水深 100[m]付近で熱水鉱床の観測に成功している(Fig.5-14)。

Aqua Box[5-9,5-25,26,27]は、2006年に九州工業大学石井研究室で開発され、主に近海域でのミッションを想定した水中ロボットである。モジュール構造によって形成されている為、ミッションによって形を変える事が可能であり。AUV、ROV両方のシステムに変更する事が可能である。日本で唯一、自律型水中ロボットの競技会AUVSIに参加しているチームである(Fig.5-15)

以上のように現在、世界中で水中ロボットの開発が行われ、自律型水中ロボットも次々と成果をあげている。またロボットの規模も高機能で小型になりつつある。これは、コンピュータなどの技術の発展に伴うものである。また、ロボットは、コンピュータの技術発展に伴い自律化へと動き足しており、ミッションの自動化が望まれてきている[5-9]。

### 5.3 水中ロボットの運動制御

水中ロボットは、水という密度の高い(空気の約 750 倍)媒質の中で行動し、3 次元的に動き回るため、その運動について以下のような特徴を持つ。

- ・ 海中ロボットの運動は前後 (Surge), 左右 (Sway), 上下 (Heave), ロール (Roll), ピッチ (Pitch), ヨー (Yaw) の 6 自由度である。
- ・ 運動モード間の相互干渉が大きい。
- ・ 水中ロボットは、魚雷のような特殊な例をのぞいて、数ノットまでの低速の運動がほとんどである。
- ・ 水からうける揚力や抗力等の非線形流体力の影響はきわめて大きく、さらに加速度に比例する力を代表する量として、付加質量を考慮しなければならない。
- ・ 水中ロボットは浮力が利用できるために、設計の自由度は大きく、定点を保持して仕事をする定点型のロボットや、長い距離を泳ぎつづける航行型のロボットが存在する。

水中ロボットの運動方程式は、ロボットが左右対称で中性浮力、重心と浮心が一致すると仮定すると次式で表現される[3],[28]。

$$\begin{aligned}
 m(\dot{u} + qw - rv) &= F_{A_x} + F_{H_x} + F_{T_x} \\
 m(\dot{v} + ru - pw) &= F_{A_y} + F_{H_y} + F_{T_y} \\
 m(\dot{w} + pv - qu) &= F_{A_z} + F_{H_z} + F_{T_z} \\
 I_{xx}\dot{p} - I_{xz}\dot{r} + (I_{zz} - I_{yy})qr - I_{xz}pq &= M_{A_x} + M_{H_x} + M_{T_x} \\
 I_{yy}\dot{q} + (I_{yy} - I_{zz})rp + I_{zx}(p^2 - r^2) &= M_{A_y} + M_{H_y} + M_{T_y} \\
 I_{zz}\dot{r} - I_{zx}\dot{p} + (I_{yy} - I_{xx})pq - I_{xz}qr &= M_{A_x} + M_{H_x} + M_{T_x}
 \end{aligned} \tag{5.1}$$

$m$ ,  $I$ は質量及び慣性モーメント,  $u$ ,  $v$ ,  $w$ は並進速度,  $p$ ,  $q$ ,  $r$ は角速度,  $F$ ,  $M$ は力及びモーメントを表す. 添え字の  $A$  は付加質量による項,  $H$  は流体力,  $T$  はロボットによる推力を意味する。

水中環境において行動する水中ロボットの運動解析[5-29,30,31]は、非線形な流体力や付加質量, 潮流に代表される外乱等のため不確定な要因が多く難しい課

題である。1980年代以降、ニューラルネットワークやファジー等、Computational Intelligence[5-32,33]と呼ばれる新しい情報処理技術が注目され、水中ロボットへも様々な応用が試みられている。

ニューラルネットワークを水中ロボットの運動制御へ応用した例は、藤井らによる運動モデルを経由して間接的にコントローラを設計する手法[5-34,35,36,37]や、須藤らによる海中ロボットの定高度航行への応用[5-38], [5-39], Yuhらが提案するロボットの実験データから直接的にコントローラを調整する手法[5-40,41]やファジーとの組合せ[5-42], Guoらによるオンライン調整法[5-43], PID 制御や非線形制御とニューラルネットワークを組み合わせた手法[5-44,45]がある。大別すると、フォワードモデルを介して間接的にコントローラを生成する手法、制御誤差から直接的にコントローラを生成する手法、既存の制御理論とニューラルネットワークの組合せに分類することができる。船舶に適用した例としては、R.S. Burns による教示型のコントローラ[5-46]や小川原らによる学習型フィードフォワードコントローラ[5-47]等がある。R.S. Burns は、シミュレーションにおいて速度を変えた場合の制御性能を最適制御と比較しており、良好な結果が得られることを示した。小川原らは、PID コントローラと学習型フィードフォワード コントローラを組合せ、風などの外乱を補償するシステムを構築している。また、山本らはスラストに変わる手法として、船のラダーを左右方向に振動させて推力を得る 手法を考案しており、推力の制御にニューラルネットワークを用いている[5-48]。

## 5.4. 目的

水中ロボットには、高い自律性の要求のために開発すべき課題が多く存在する。例えば、複雑な動特性、環境認識の必要性、行動決定手法の確立、自己一同定などが挙げられる。本論文では、知能化のアプリケーションの1つとして水中ロボットの運動制御について扱い、様々なプラットフォームや環境下に適応できる制御システムの開発を目標とする。

水中ロボットの運動制御システムは、以下に起因する水中ロボットの動特性の変化に影響を受ける。

- ・ ミッションによる搭載機器の変更
- ・ マニピュレータの位置及び姿勢
- ・ スラスタの特性
- ・ プラットフォーム
- ・ 用流や波等の外乱

このような動特性の変化を想定して、様々な状況に適応できる制御器が必要となる。水中ロボットの制御器は、非線形かつ複雑な動特性を考慮した制御システムを構築しなければならない。つまり、動特性の変化やプラットフォームの変更に対して、ロボット自身が適応的に制御器を変更する事が出来れば望ましい。このような問題には、階層型ニューラルネットワークに代表される教師あり学習が有効である。現在までにニューラルネットワークをはじめとする脳型情報処理アルゴリズムの仕組みを参考にした学習能力に本研究では注目し、水中ロボットの運動制御に関しての研究が本研究室では行われてきた。使用するプラットフォームには非潜航型の自律型水中ロボットTwin-Burger(Fig.5-16)を用いられた。石井らは、水中ロボットの動特性をリカレント型ニューラルネットワークで表現し、オンラインで制御器と動特性の調整を行っている[49-52]。石井らによって提案されたニューラルネットワークによる適応制御手法では、水中ロボットの動特性及び制御器をオンラインで獲得することができるが、一対の動特性と制御器しか持っていないために適応する(動特性と制御器を調整する)過程で、過去における学習によって得た情報の影響は低下していく。つまり、過去に

において適応していた動特性及び環境に戻った場合においても再学習する必要があり、常時、ロボットにおいて計測される時系列情報を追加学習しなければならない。より高速に、かつ柔軟に動特性などの変化に適応するには、過去を含めた様々な適応状態を保持しながら、新しい動特性や環境へ適応する行動獲得システムが必要となる。つまり、ロボットが状態や環境と行動の関係を記憶して保持しながら学習によって自己組織的に得る事が望ましいと考えられる。そこで、本研究では、自律型水中ロボットにおける自己組織的行動獲得システムの確立と環境にロバストなセンシングシステムの開発を目標とする。そこで、パターン認識のひとつの手法として自己組織化マップ(SOM)を用いると仮定する。しかしながら、一般的な自己組織化マップ(SOM)は入力ベクトル空間しか取り扱うことができない。さらに学習過程が教師無し学習であるため、ロボットの時系列情報から入出力関係を学習してロボットの動特性や制御器を表現するには、競合層における(入力, 出力)の組としてベクトル表現する等の工夫が必要となる。入出力の写像関係を得るには、教師有り学習アルゴリズムを持ったニューラルネットワークが適している。ロボットの行動決定システムには、ロボットがおかれた環境を自動的に認識し把握する“教師無し”アルゴリズム及び動特性や制御器獲得のための“教師有り”アルゴリズムの両者の考え方が必要であり、両者の長所を取り入れた情報処理技術が望まれる。

徳永らによって提案されたモジュラネットワーク自己組織化マップ(mnSOM: modular network Self-Organizing Map) [5-53] は、関数をモジュールの要素として扱えるため、自己組織化マップの教師無しアルゴリズムを継承しつつ、モジュールの要素として教師有り学習アルゴリズムを有する情報処理システムを導入することにより、上記の両者のアルゴリズムを同時に実現することが可能である。このような特徴を活かして、動的システムのマッピング、非線形主成分解析、自己組織化制御器などの適応例が報告されている [5-54,55,56,57]。

よって、mnSOMを用いる事で、実際に適応制御を実現する。西田によると、シミュレーション上においてmnSOMを用いる事でこの適応制御を実現している [5-58,59]。つまり提案するアルゴリズムは、Fig. 5-17に示すように、ロボットは作業フィールドの環境特徴地(Fig. 5-17 左)をセンサ情報を基に作成する。自状態と環境特徴地図を基に、目標となる経路や行動を決定する。一方、ロボットは移動しながら自状態の変化を動特性に関する知識を獲得し、対応した制御器を作成する。自状態と動特性に関する知識とを照らし合せて適切な制御器を選択す

る. 目標行動に適した制御器を選択し目標値に対して最適な出力を算出する. この出力値をロボットの制御力として採用する. 本研究では, 西田の提案したアルゴリズムを改良し, 実機Twin-Burgerを用いた適応制御システムの実現を目標とする.

## 5.5 自律型水中ロボット”Twin-Burger”

本研究で用いる水中ロボット “Twin-Burger”は, 1992年に東京大学生産技術研究所によって開発された非航行型自律型水中ロボットである. システムの老朽化と度重なるシステムの変更に伴い, 内部構造が煩雑になった為, モジュール化の手法を用いて新しくシステムを変更した[5-61,62].

Twin-Burgerは, 全長1.3 [m], 乾燥重量 110 [kg]で, バッテリーを格納するアルミニウム製シリンダ, 2つのFRP耐圧容器及びその他の機器をフレームに取り付けたオープンフレーム構造のロボットである. フレームの上部にFRP容器を株にシリンダを取り付ける事で, 重心位置と浮心位置の距離を十分にとり, ピッチとロールに関する静的安定を確保している. Table 5-1にTwin-Burgerの仕様を示す. 本研究では, このプラットフォームを参考に制御システムを作成する.

## 5.6 mnSOM を用いた運動の同定と制御

本研究では、環境やロボットの動特性の変化に応じて自己組織的にロボットの行動を獲得するシステムの開発を目標としている。先行研究として西田は、その第一段階として、動特性の変化に適応する制御システムの開発及びシミュレーションを行ってきた[5-63,64,65,66,67,68,69,70,71,72,73]。そして、先行研究をもとに本研究では、アルゴリズムの改良に取り組んできた[5-74,75,76]。

本システムが良好な制御性能を得るには、動特性の変化に素早く追従し、制御則を適応的に調整していく必要がある。そこで、使用したアルゴリズムは、“教師無し”及び“教師有り”アルゴリズムの長所を取り入れたモジュラネットワーク自己組織化マップ(mnSOM: modular network Self-Organizing Map)を用いて実現した。ロボットの制御において扱う情報は、時系列データであるのでモジュールにMLPのひとつであるリカレント型のニューラルネットワークを用い、mnSOMの基本モジュールとする。学習アルゴリズムは、2章で述べたMLP型mnSOMのアルゴリズムと同じである。リカレントニューラルネットワーク型mnSOMを用いて、状態量と操作量に関する様々な時系列データを学習させることにより、ロボットの動特性を取得しながら、異なった動特性を有するモジュール群を競合層に配置していく。本研究で提案する制御システムは、Fig. 4-17,18示すように、ロボットの動特性を表現するフォワードモデルマップ及びこれに対応したコントローラマップの2層のリカレントニューラルネットワークから構成されている。

リカレントニューラルネットワーク型mnSOMを用いた適応型制御システムは、フォワードモデルマップの作成、コントローラマップの作成、及びロボット制御への適用の3つのフェーズ構成されている。各フェーズに関して以下に述べる。

## 5.6.1 適応制御システム

### 5.6.1.1 フォワードモデルマップの作成

動作環境の変化や動特性の変動, アクチュエータの特性変化や故障を想定するか, 実機により計測し, 状態量と操作量の組からなるデータクラスを準備し学習する. Fig5-18-(a)において, ある動特性を表現する時系列データ  $S_i(\mathbf{x})$  は矢印で示された赤色の競合層モジュールが勝者モジュールとして対応し, 別の動特性を表現する時系列データ  $S_j(\mathbf{x})$  も同様に矢印で示された競合層モジュールが勝者モジュールとして対応している状況を示している. モジュールの色は学習した動特性の影響を意味しており, 学習の過程で, 勝者モジュールの間には  $S_i(\mathbf{x})$  と  $S_j(\mathbf{x})$  から得られた動特性によって補間された動特性を有するモジュールが得られる. 各モジュールの学習係数及び学習回数は, 同じとして各データクラスに対する勝者モジュールが選択され, 学習が収束したと判断された段階で学習を終了し, 次の学習段階に移行する. 得られた学習結果は, 以降, フォワードモデルマップと呼ぶ.

水中ロボットの速度及び制御入力と加速度の関係を得るために, リカレントニューラルネットワーク (Fig5-17) においてフォワードモデルとして示したネットワーク) を基本モジュールとしたmnSOMを用いてフォワードモデルマップを作成する. 以降, 状態変数  $S$  は位置,  $\Delta S$  は速度及び  $\Delta^2 S$  は加速度,  $u$  は制御入力を示す. フォワードモデルマップ作成のアルゴリズムを以下に述べる.

$M$  個の時系列データクラス  $D_i = (D_1, D_2, \dots, D_M)$  があり, それぞれ  $N$  単位時間の入出力対  $(\mathbf{x}_{ij}, \mathbf{y}_{ij}) = \{(\mathbf{x}_{i1}, \mathbf{y}_{i1}), (\mathbf{x}_{i2}, \mathbf{y}_{i2}), \dots, (\mathbf{x}_{iN}, \mathbf{y}_{iN})\}$  を持っており, 関数  $f_i(\cdot)$  を構成しているとする, 式(5.2)の関係が成り立つ.

$$D_i = (\mathbf{x}_{ij}, \mathbf{y}_{ij}) = \{(\Delta S_{ij}, u_{ij}), \Delta S_{ij}\} \quad (i=1 \sim M, j=1 \sim N)$$

$$\Delta S_{ij} = f_i(\Delta S_{ij}, u_{ij}) \quad (5.2)$$

一方, mnSOM のモジュールが  $K$  個あり,  $k$  番目のフォワードモデルの持つニューラルネットワークの結合加重は  $w_k$  であるとする, 入力データ  $(\Delta S_{ij}, u_{ij})$  と出力データ  $\Delta^2 \hat{S}_{ij}^k$  は, 式(5.3)に示す関数  $F_k(\cdot)$  を構成している.

$$\Delta^2 \hat{S}_{ij}^k = F(\Delta S_{ij}, u_{ij}; w^k) = F^k(\Delta S_{ij}, u_{ij}) \quad (5.3)$$

このとき, 学習は以下のように行われる.

i) 初期化: mnSOM の結合加重  $w_k$  を乱数で初期化する

ii) 評価過程: 式(5.4)を用いて,  $\Delta S_{ij}(k)$  と  $\Delta 2S_{ij}$  の平均二乗誤差  $E_{Fwd_i}$  のクラス  $D_i$  に対して計算する. 式(5.8)は  $f_i(\cdot)$  と  $F_k(\cdot)$  の距離を求めていることになる.

$$E_{Fwd_i}(k) = \frac{1}{2N} \sum_{j=1}^N \left\{ \Delta \hat{S}_{ij}(k) - \Delta S_{ij} \right\}^2 \quad (5.4)$$

iii) 競合過程: 式(5.5)に従って, 誤差が最小となる勝者モジュールのインデックスを  $k_i^*$  を決定する.

$$k_i^* = \arg_k \min(E_{Fwd_i^k}) \quad (5.5)$$

iv) 協調過程: 式(5.6)に従って勝者モジュールからの距離および学習回数によって, 学習分配率  $\Psi_{ik}$  を決定する.  $\Psi_{ik}$  は  $k$  番目のモジュールが  $i$  番目のデータクラスの入出力関係を学習する量を示す.  $h(\cdot)$  は近傍関数で, 距離  $L$  及び学習回数  $T$  が増加するにつれて単調減少する関数を選択する.

$$\Psi_{ik}^k = h\{L(k, k_i^*), T\} / \sum_{\tilde{i}=1}^M h\{L(k, k_{\tilde{i}}^*), T\} \quad (5.6)$$

v) 適応過程: 誤差逆伝播法によって学習する. 式(5.7)に示すように学習率  $\eta$  と学習分配率  $\Psi_{ik}$  を用いて  $w_k$  を更新する.

$$\Delta w^k = -\eta \sum_{i=1}^M \left\{ \Psi_{ik}^k \frac{\partial E_i^k}{\partial w^k} \right\} \quad (5.7)$$

ii) ~ v) の過程を学習が収束するまで繰り返す.

リカレントニューラルネットワーク型 mnSOM を用いた場合でも基本的に mnSOM アルゴリズムには変更はないが, 入力データとして与える水中ロボットの時系列データは, 順序に意味があるのでバッチ型の学習法を用いる. ここで, 西田の提案するアルゴリズムでは, 出力の  $\Delta S$  を入力に学習の 1 回目から入力

値に返している。その為、学習が上手く十分されずに速度の値を返すと収束しなかった。そこで、本手法では、十分に学習をした後に出力値を返すようにアルゴリズムを変更した。

### 5.6.1.2 コントローラモデルの作成

第2段階として、作成したフォワードモデルマップを用い、各モデルに対応したコントローラを作成する。Fig 5-18(b)の上部に示されたコントローラマップの作成において、競合層における幾何学的な配置は考慮せず、対応するフォワードモデルに対してコントローラを適応させるものとする。コントローラマップにおける各コントローラに対して制御目標値が与えられる。対応するフォワードモデルから得られるロボットの状態量と目標値との偏差量をもとに操作量が算出され、フォワードモデルマップに送信される。フォワードモデルマップでは、操作量から計算される次の時間ステップの状態量と目標値との誤差をコントローラマップに逆伝播することにより、誤差逆伝播法を用いて制御則の調整を行う。制御目標値、学習係数及び学習回数ここではすべてのコントローラにおいて同じ値とし、学習が収束するまで行う。

フォワードモデルマップのときと同様に、リカレントニューラルネットワーク型mnSOMを用いて、目標値と現在の状態の誤差から制御入力を決定するコントローラマップを作成する。Fig5-18-(b)に示すように、初期コントローラマップと学習済みのフォワードモデルマップを結合し、Fig5-17に示したコントローラとフォワードモデルのネットワークを一つのネットワークとした基本モジュールを持つmnSOMとして構成し、フォワードモデルの各ウェイトを固定して、コントローラのウェイトを調整する。以下にアルゴリズムを述べる。

コントローラマップ作成における入力データクラス $D_i$ は目標状態 $r = (r, \Delta r)$ を用いて以下の式(5.8)のように表現する。

$$\begin{aligned} D_i = (x_{ij}, y_{ij}) &= \left\{ (r_{ij} - S_{ij}, \Delta r_{ij} - \Delta S_{ij}, u_{ij-1}), u_{ij} \right\} \\ u_{ij} &= g_i(r_{ij} - S_{ij}, \Delta r_{ij} - \Delta S_{ij}, u_{ij-1}) \end{aligned} \quad (5.8)$$

ここで、 $r$  は目標位置、 $\Delta r$  は目標速度である。

一方,  $k$  番目のコントローラの持つネットワークの結合加重は  $v_k$  と表すとすると, 入力  $(r_{ij} - S_{ij}, \Delta r_{ij} - \Delta S_{ij}, u_{ij-1})$  と出力  $u_{ij}$  は, 式(5.9)に示す関数  $G_k(\cdot)$  を構成している.

$$\begin{aligned}\hat{u}_{ij}^k &= G(r_{ij} - S_{ij}, \Delta r_{ij} - \Delta S_{ij}, u_{ij-1}; v^k) \\ &= G^k(r_{ij} - S_{ij}, \Delta r_{ij} - \Delta S_{ij}, u_{ij-1})\end{aligned}\quad (5.9)$$

先に述べたように, mnSOM の学習のプロセスに変更はないが, コントローラマップ作成において異なるのは, 評価過程における評価関数と, それに伴って変化する適応過程における結合加重の更新式である. 更新式そのものは式(4.7)と同じである. 式 (5.10)にコントローラマップに対する評価関数を示す. 式(4.10)は, 目標状態に現在の状態が近ければ, また, 制御入力の絶対値が少ないほど, 値が小さくなり, 良い評価であることを示している.

$$E_{cli}^k = \frac{1}{2N} \sum_{j=1}^N \left\{ p_1 (r_{ij} - S_{ij})^2 + p_2 (\Delta r_{ij} - \Delta S_{ij})^2 + p_3 u_{ij}^2 \right\} \quad (5.10)$$

ここで,  $p_1, p_2, p_3$  は, それぞれ位置, 速度及び制御入力に対する評価値への重み係数である.

### 5.6.1.3 ロボットへの適応

第1段階及び第2段階の学習は, 準備されたデータのオフライン学習及び調整であった. 第3段階では制御対象であるロボットの制御と並行して, ロボットの動作環境や制御特性の変化に対してオンライン学習による適応を行う. ロボットのミッション遂行中において, ある一定期間サンプリングされた時系列データ (Fig4.18-(c))における  $S_i(\mathbf{x})$  をフォワードモデルマップへ入力する. フォワードモデルマップは動特性推定器として用いられ, 最も動特性を表現する勝者モジュールが選択される. ロボットの制御には, 勝者となったフォワードモデルに対応するコントローラから得られる操 量がロボットへの制御入力として採用される. また, これと並行して, フォワードモデルマップの追加学習による動特性

変化への適応, 及びフォワードモデルマップの更新に伴うコントローラマップの調整が行われる. フォワードモデルマップを用いた動特性の推定と制御入力の決定は以下のように行う.

$N$ 時間観測したデータを用いて, 式(5.11)を用いてフィードフォワードモデルとの誤差を計算する.

$$E_{Exp}^k = \frac{1}{2N} \sum_{j=1}^N \left\{ \Delta \hat{S}_{Exp_j} - \Delta S_{Exp_j} \right\}^2 \quad (5.11)$$

次に, 式(5.11)に従って, 観測された状態を最も良く表現しているフォワードモデルのインデックス  $k_{Exp_j}^*$  を求める.

$$k_{Exp_j}^* = \arg_k \min(E_{Exp}^k) \quad (5.12)$$

インデックス  $k_{Exp_j}^*$  に対応するコントローラの出力をロボットへの制御入力として採用する.

$$\mathbf{u}_{Exp} = \mathbf{G}^{k_{Exp_j}^*} (\mathbf{r}_j - \mathbf{S}_{Exp_j} \Delta \mathbf{r}_j - \Delta \mathbf{S}_{Exp_j} \mathbf{u}_{Exp_{j-1}}) \quad (5.13)$$

## 4.6.2 シミュレーション

### 4.6.2.1 フォワードモデルマップのシミュレーション 及び考察

提案手法の第1段階として, Yaw方向のフォワードモデルマップ作成シミュレーションを行った. 入力時系列群の関係とそれらを補間した動特性が反映されたマップが得られるかを検証するために, まずは, ひとつの運動モードに関してパラメータを変えてシミュレーションを行った.

式(5.14)に単純化した水中ロボットの運動方程式を示す.

$$F = M\ddot{x} + C\dot{x}|\dot{x}| \quad (5.14)$$

$F$  は力,  $\dot{x}$  は速度,  $\ddot{x}$  加速度を示しており,  $M, C$  はそれぞれ質量及び付加質量, 非線形流体力に関するパラメータである. 式(5.14)を用いてパラメータ  $M, C$  を Table 5-2に示すように変化させ, 速度の絶対値が45.0 [deg]を超えたら制御入力 5.0 [N]の方向を反転させるリミットサイクル運動の時系列データを50秒間, 10[Hz]でサンプリングしたものを用意し, 10x10格子状モジュール構造の mnSOM に入力した.

Fig. 5-19は, Table 5-2を学習データとして提示したときの特徴マップを示している. 16個のパラメータから生成した時系列をmnSOMへ入力し, 500,000 回学習を行った結果の得られたマップである. マップの各正方形は フォワードモデルを表しており, それぞれある動特性を表現したリカレントニューラルネットワークを持っている. 入力時系列に対する勝者モジュールにはその時系列の速度(青細線), 加速度(赤太線)を, 時間を横軸にとってプロットしている. 入力したデータクラス  $D_i (i = 0 \sim 15)$  は, それぞれマップ上の(1,4), (1,6), (2,7), (2,9), (2,1), (4, 4), (5,5), (6,7), (5,1), (7,1), (7,3), (8,5), (10, 1), (9,1), (10, 3) and (10, 4)に配置されている. 得られたフォワードモデルマップを検証するため, 各モジュールに対してリミットサイクル試験を行った. 各モジュールから得られた時系列データを用いて, 最小二乗法によって運動方程式のパラメータを算出して, 各モジュールが表現している動特性を数値的に評価する. 運動方程式の係数  $M, C$ の推定を行った結果をFig.5-20に示す.

Fig5-20における青い四角は入力データ作成時に用いた係数, 赤いマークはフォワードモデルマップから推定された各フォワードモデルモジュールのパラメータを表している. パラメータ  $M-C$  空間において, フォワードモデルマップは格子状に広がっており, 入力データのパラメータに完全には一致していないものの, 中間に存在するモジュールは入力データによって補完された中間的な振る舞いを表現していることがわかる.

### 5.6.2.2 コントローラモデルマップのシミュレーション及び考察

得られたフォワードモデルマップを用いる事でコントローラマップを作成する. コントローラマップは, Fig.5-17に示すように, 3層のニューラルネットワークで構成されている. コントロールマップの結合加重は, ニューラルネットワークのモデルとして, PIDコントローラにおける比例ゲイン, 積分項ゲイン, 微分項のゲインをすべて1.0と設定して, 入力を $[-1.0, 1.0]$ を0.1刻みずつ与え, 出力結果を学習したものを初期の結合加重とする. その後, 設定した目標軌道であるにフォワードモジュールの出力が追従するようにコントローラの結合加重を調整する. フォワードモデル, コントローラモデルの両者を一つのMLPとして考え, フォワードモデルから出力される状態量を5.10式によって評価して, 誤差を誤差逆伝搬法によって学習させる. このとき, 5.10式に用いられるゲイン $P1, P2, P3$ はそれぞれ $P1 = 1.0, P2 = 8.0, P3 = 0.0$ と設定して実験を行った. なお, このときフォワードモデルとコントローラモデルの関係は1対1の関係となっており, 1つ1つのモジュールに対して学習を行っている.

コントローラマップ生成の為, 目標の位置を $0 \sim 25$  [sec]のとき $\pi/4.0$  [rad],  $25 \sim 50$  [sec]のとき $-\pi/4$ と設定し, 目標速度を $0.0$  [rad/sec]とした.  $10$  [Hz]でサンプリングしたときの時系列をすべてのコントローラマップに対して目標軌道として入力した. このときの各モジュールの学習結果をFig.5-21に示す. モジュールは $10 \times 10$ のモジュールである. 図は, (行の番号)  $\times$  (列の番号) を示しており, 各モジュールは, Fig. 5-19のモジュールに対応している. 図では, 横軸は時間[sec]を示しており,  $0 \sim 50$  [sec]までの結果を示している. 縦軸は, 力, 速度, 位置を示しており, それぞれ,  $[-5.0, +5.0]$ の値を $[-1.0, +1.0]$ の値にノーマライズして, 表示している. 青線は, 力の推移を表しており, 水色の線は, 細線が目標とな

る位置を示している。また、水色太線は、位置の推移を示している。最後に、赤色の線は速度の推移を示している。すべてのモジュールにおいて目標の位置に追従しようとしている。しかし、力がかなり振動している。そのときの速度の変化はほとんどないように伺える。つまり、力を出しているが速度の変化は非常に少ない。今回の学習に与えた時系列のデータは、 $-5.0$  or  $+5.0$ [Nm]の力という力しか操作量として与えていない。しかし、コントローラとして様々な力を推定する為には、様々な力を与えた学習データを学習した、フォワードモデルに改良する必要があると考えられる。また、評価関数について考察すると力の差分について考慮する項目が必要となるのではないかと考えられる。力の差分項は、実質力の影響に大きく影響すると考えられる。そして、本実験においてシミュレーション中に非常に評価関数のパラメータを調整する事が困難であった。そこで、今後の展望としてこの評価関数のパラメータの最適化アルゴリズムがあるとより設計が容易になると考えられる。

#### 5.6.2.4 実機性能実験および考察

Twin-Burgeを用いた実機実験を行う。Twin-BurgerのYaw方向の運動方程式は、5.14式によって示される。このとき、首藤らによって実験された基礎性能実験によって $M=67.9$ ,  $C=196$ と推定されている[5-77]。しかし、藤井らによって示された基礎実験では $M=29.3$ ,  $C=1.99$ と推定されている[5-78]。これらの違いは、システムの改良などや搭載部品等の変更によっておこったものであると考えられるが、今回は、フォワードモデルにセンサからの情報を取り出し、その速度とフォワードモデルから推定された速度との差が一番近いモジュールをベストマッチングモジュールとする。実験の結果、Fig. 5-19における3行目10列目のモジュールがベストマッチングモジュールとして選定された。3行10列目のモジュールは、フォワードモデルから5.14式の $M, C$ を推定したところ、 $M=103$ ,  $C=41.3$ を示している。mnSOMの性質上学習時系列データの $M, C$ の内装部分しか想定できないためこのモジュールがベストマッチングモジュールとして選ばれた。想定結果から現在のシステムでは首藤らによって推定された結果にダイナミクスが近い事が考えられる。

次に、3行10列目のコントローラモジュールを用いて制御実験を行った。そのときの制御実験の結果をFig.5-22に示す。目標に対して、追従している事がわか

る。しかし、力の推移を見ると、目標の位置に到達した後も大きく力が発生している事がわかる。実際にFig.5-21で示したコントローラの学習に置いても定常状態において力が大きく触れている事がわかる。つまり、位置が定常状態になった後のコントローラの調整が不十分であると考えられる。

ここで、コントローラマップの再学習を行ったが、今回のフォワードモデルを用いた学習では、これ以上良いコントローラを作成する事は出来なかった。それには、問題点が挙げられると考えられる。それは、フォワードモデルの学習には、ある一定の力（今回の場合、 $-5.0$  [Nm] or  $+5.0$  [Nm]）しか、学習のときに入力として与えておらず、それ以外の力を出すときのフォワードモデルが十分に推定できていない部分がある。Fig. 5-12 の06x05番目のモジュールを代表としてあげると、定常状態となった後にも、力が出ているが速度に変化がない事が挙げられる。これは、フォワードモデルの出力として、力が小さいときは、ほぼ出力となる加速度が $0.0$ を示している事が読み取れる。つまり、 $[-5.0, +5.0]$  [Nm]での力を制御したい場合、この間の力を様々な利用した運動の入出力関係を学習させる必要があると考えられる。

本稿では、ロボットがコントローラモデルを用いて目標に対して制御をする可能性を示す事が出来た。コントローラモデルの評価関数のパラメータ調整やフォワードモデルの再構築が将来の実装にたいする展望として考えられる。

## 5.8 まとめ

本章では、mnSOMを用いた水中ロボットの運動の同定と制御システムの開発を行った。動特性の異なる水中ロボットの運動時系列をリカレント型ニューラルネットワークmnSOMを採用して学習する事で、同時に複数の動特性を表現したリカレントニューラルネットワーク（フォワードモデル）を得るとともに、入力時系列間の関係を学習する事が出来た。同時に、フォワードモデルマップ内に入力時系列群の中間状態を表現したネットワークを獲得する事も出来た。フォワードモデルマップに対応した複数の制御器をフォワードモデルの出力を用いて、フィードバック系のニューラルネットワークモデルとして、コントローラモデルを作成した。獲得したフォワードモデルとコントローラマップを用いた制御手法を自律型水中ロボットを用いてその有効性を確かめた。今回の実験では有用性を確かめる事が出来たが、より精度の高い制御を行う為にはいくつかの改良が必要である。一つは、評価関数のパラメータ調整である。パラメータ調整に関しては、ニューラルネットワークで力を推定しているのと同様に自動的に最適なパラメータを推定するようなシステムが必要である。そこで将来の展望として、そのパラメータを推定する為のコントローラモデルを1層追加したモデルを考えればよいのではないかと考えられる。二つ目は、フォワードモデルの再構築である。フォワードモデルは、ダイナミクス推定を行う為、様々な航行を行ったときの時系列データを入れる必要がある。現在は、リミットサイクル実験での結果のみであったが、将来の展望としては、実機を用いた様々な航行データを学習データとすることでより良いフォワードモデルが構築できるのではないかと考えられる。

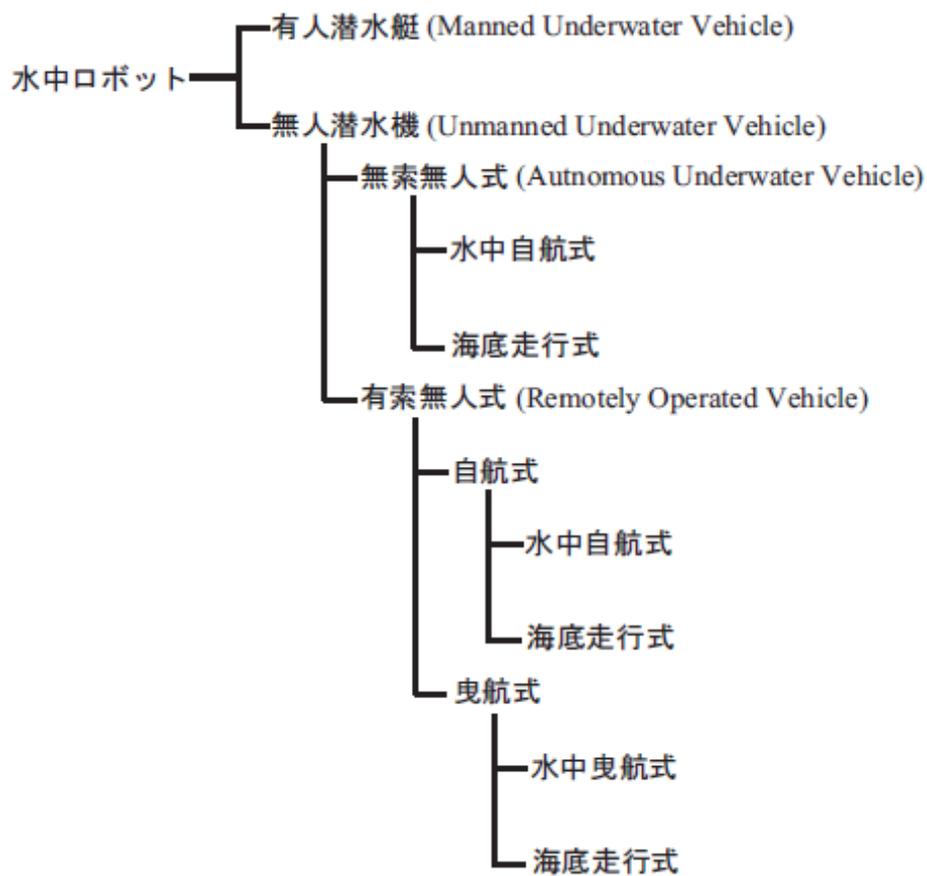


Fig.5-1 Classification of underwater robots [5-19 参照]



Fig.5-2 ALVIN ([http://www.whoi.edu/marops/vehicles/alvin/index\\_en8.html](http://www.whoi.edu/marops/vehicles/alvin/index_en8.html))



Fig.5-3 ABE

(<http://www.whoi.edu/sbl/liteSite.do?litesiteid=4050&articleId=6343>)



Fig. 5-4 Odyssey II

(<http://auvlab.mit.edu/vehicles/vehiclespec2x.html>)

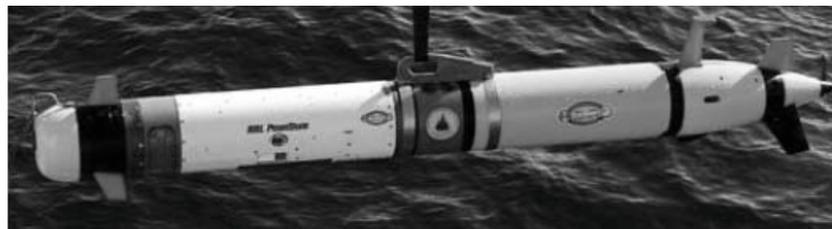


Fig.5-5 REMUS600

(<http://www.whoi.edu/sbl/image.do?id=22227&litesiteid=7212&articleId=11313>)



Fig.5-6 Jason

(<http://www.whoi.edu/marops/vehicles/jason/index.html>)



Fig.5-7 HYSUB5000

(<http://www.ise.bc.ca/hysub5000j.html>)



Fig.5-8 HEMIRE  
(KORDI,2005 Annual Report より)



Fig.5-9 しんかい 6500  
(<http://www.jamstec.go.jp/jamstec-j/gallery/yujin/6500.html>)



Fig.5-10 かいこう(亡失以前)

(<http://www.jamstec.go.jp/jamstec-j/gallery/mujin/kaiko.html>)



Fig.5-11 R-One Robot

(<http://underwater.iis.u-tokyo.ac.jp/>)



Fig.5-12 r2D4  
(<http://underwater.iis.u-tokyo.ac.jp/>)



Fig.5-13 AquaExplorer2000  
(<http://www.k-kcs.jp/japanese/aqua2000-j.html>)

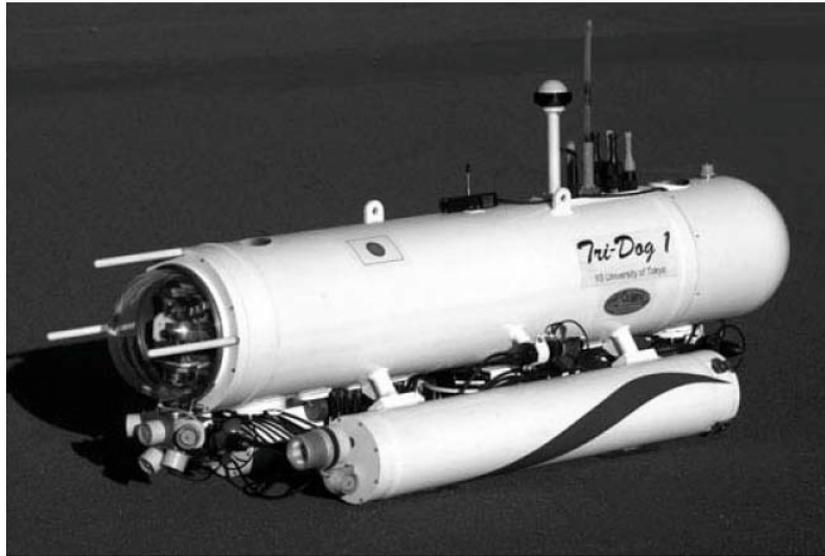


Fig.5-14 Tri-Dog 1  
(<http://underwater.iis.u-tokyo.ac.jp/>)

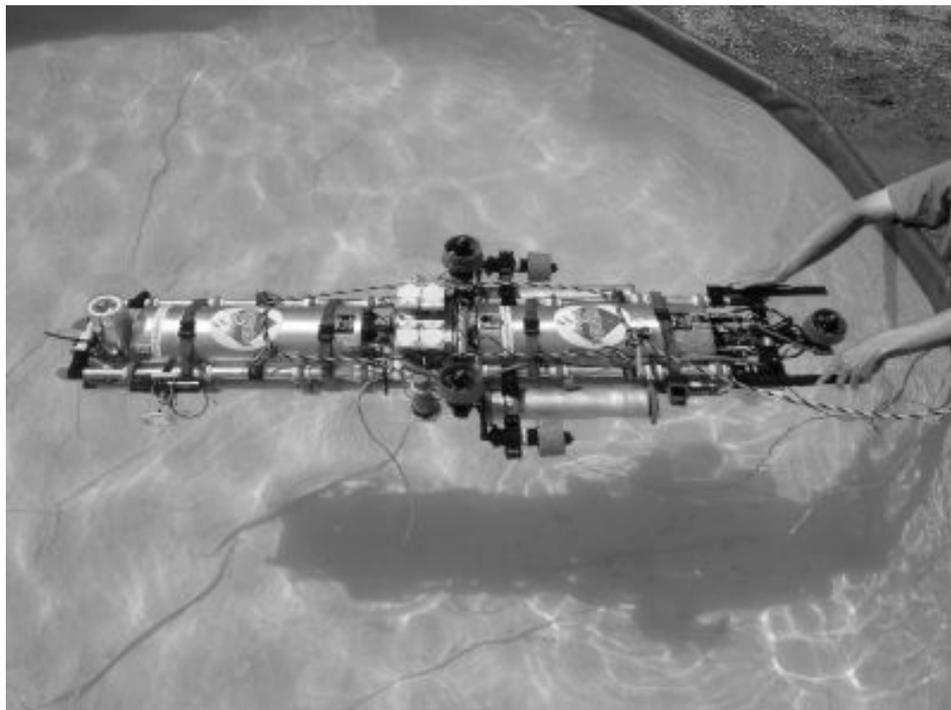


Fig. 5-15 Aqua-Box III  
([http:// www.brain.kyutech.ac.jp/~ishii/index\\_jp.html](http://www.brain.kyutech.ac.jp/~ishii/index_jp.html))

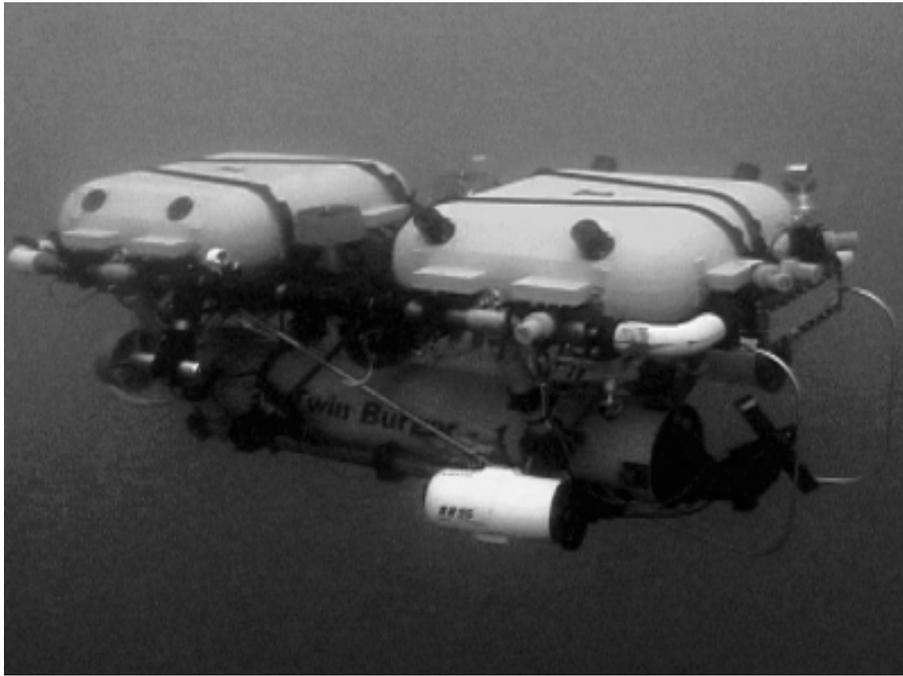


Fig. 5-16 AUV “Twin-Burger”

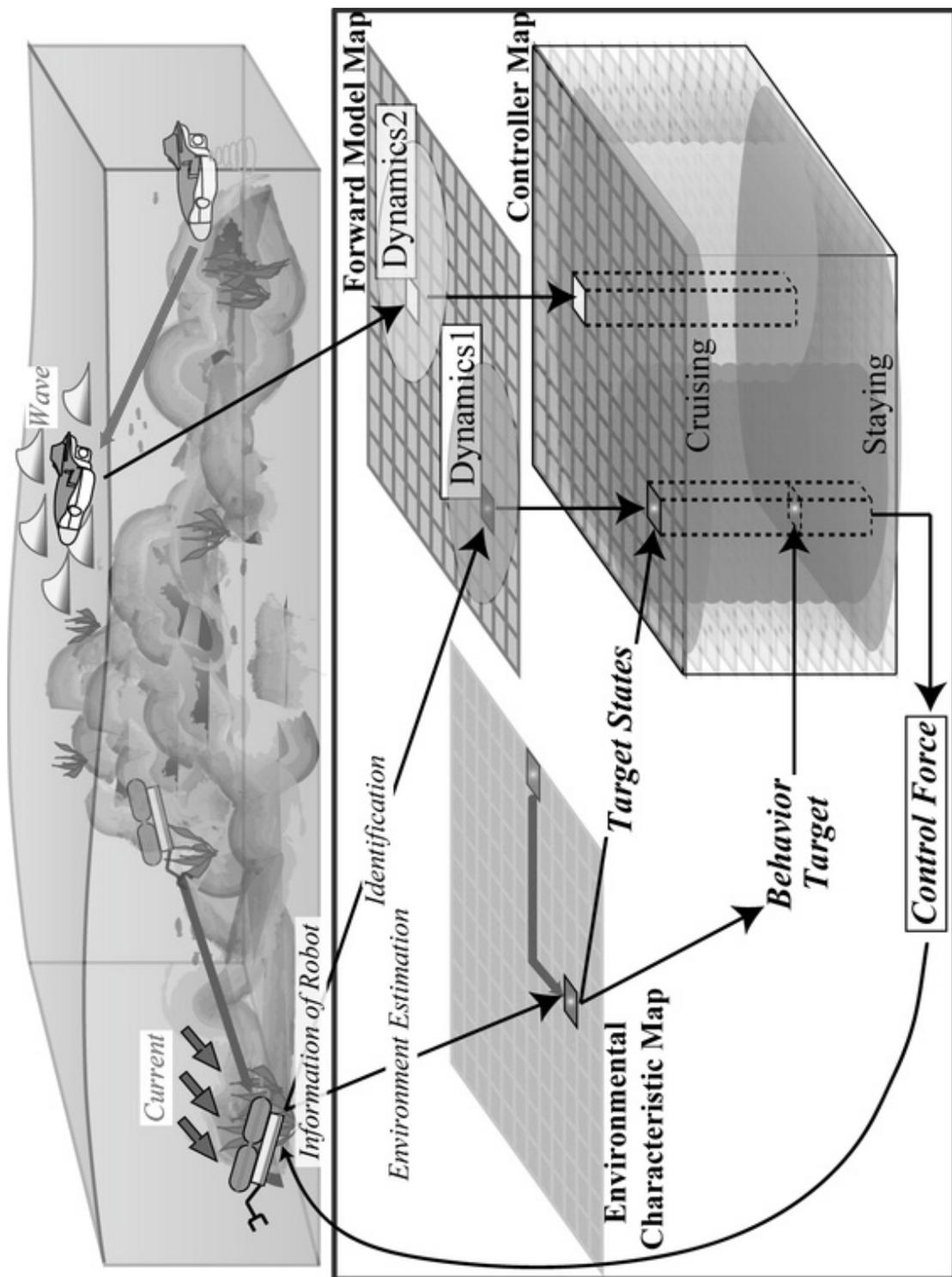


Fig. 5-16 Self-Organizing Decision Making System for AUVs[59]

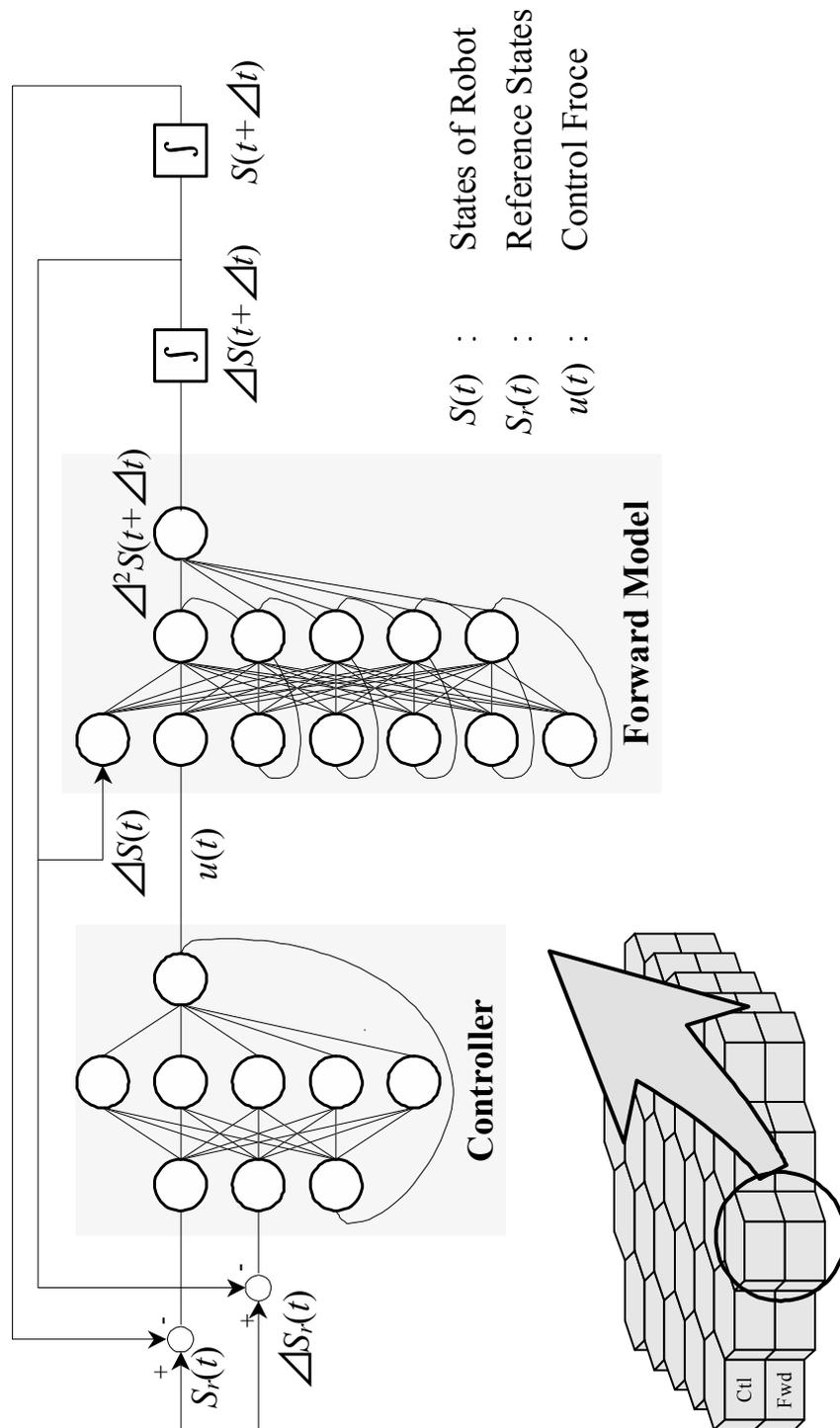
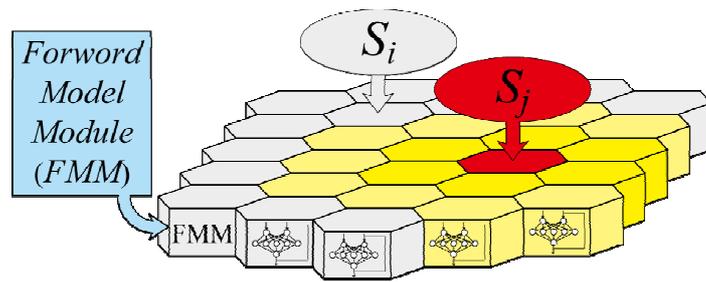
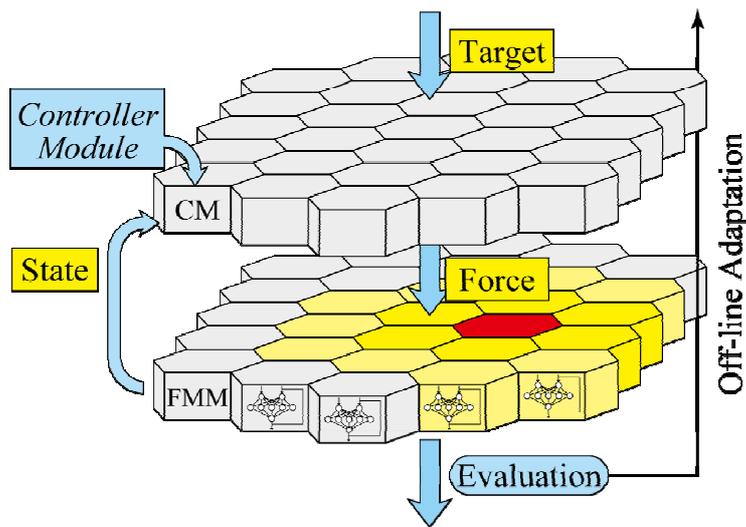


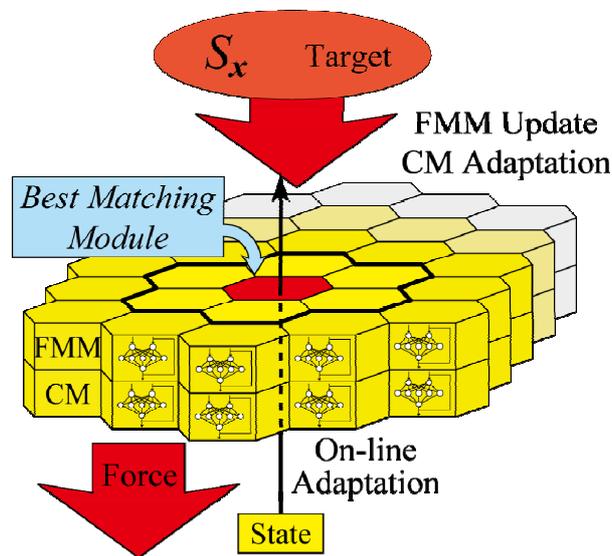
Fig. 5-17 The Network Structure of a Forward Module and a Control Module



(a) Building a Forward Model Map



(b) Adaptation of a Controller Map using the Forward Model Map



(c) Implementation of the Control Map to Robot Control and On-line Adaptation using the Forward Model Map and the Controller Map

Fig. 5-18 Learning Processes of an Adaptive Controller System using RNN-mnSOM (RNN: Recurrent Neural Network)

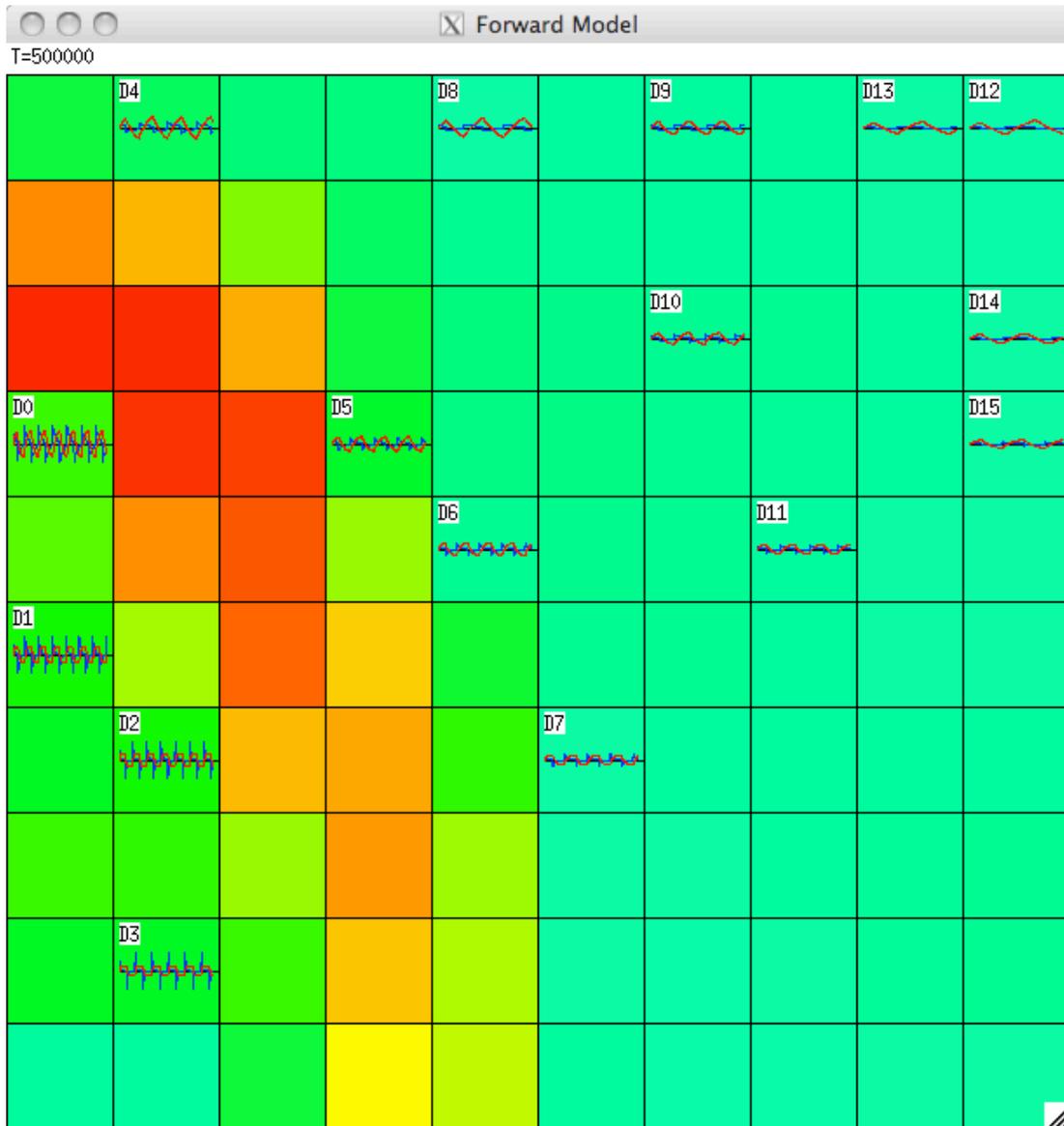


Fig. 5-19 A Forward Model Map Obtained from the Time Series of limit Cycle Simulation Data

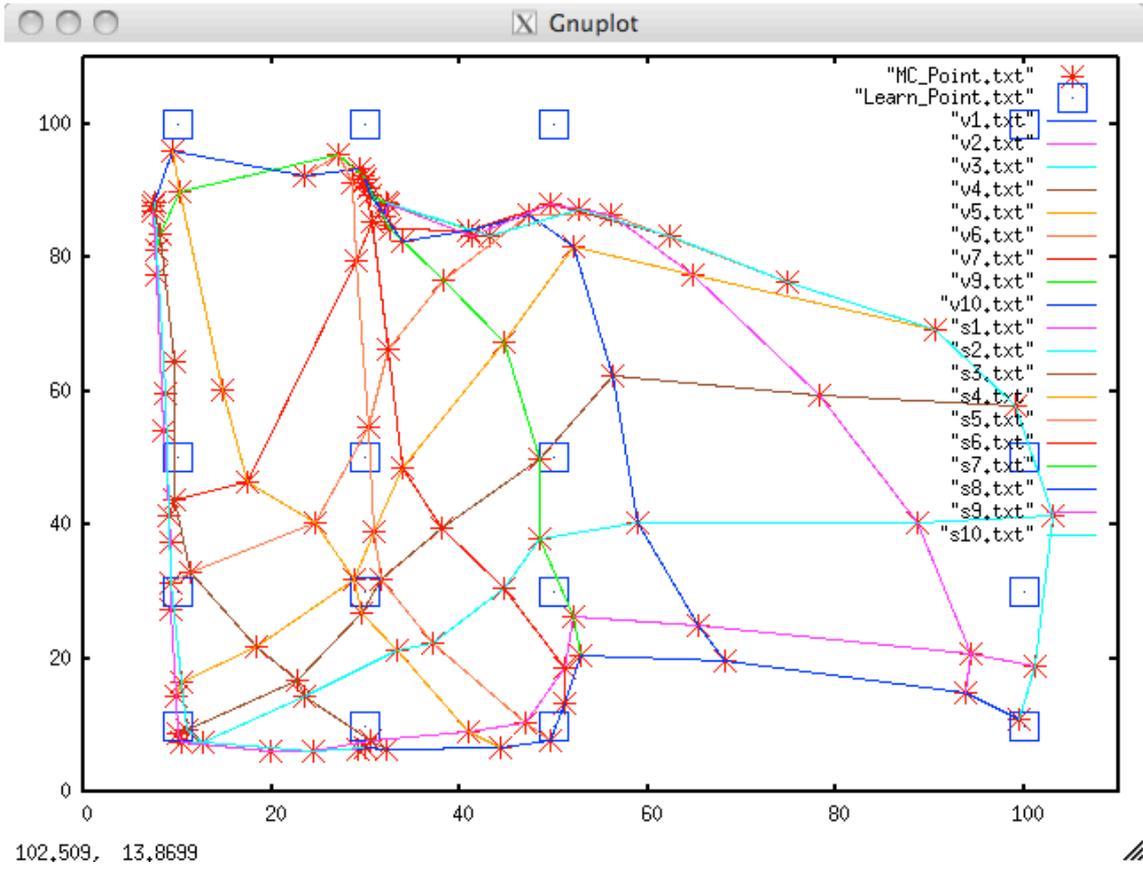
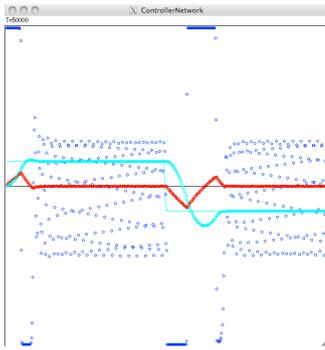
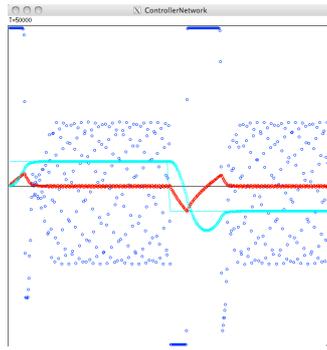


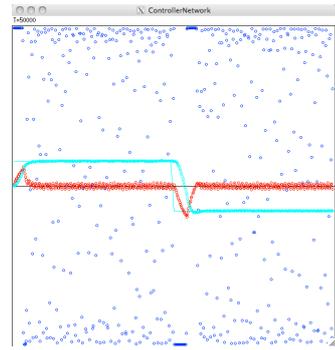
Fig. 5-20 Forward Model Map Evaluation in M-C Space calculated by the Means Square Method



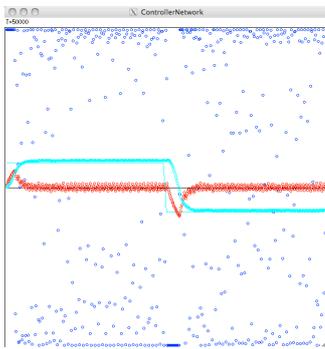
01 x 01



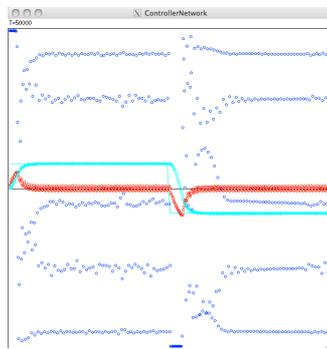
01 x 02



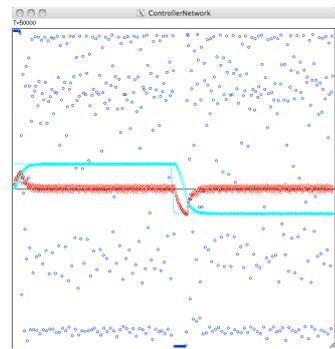
01 x 03



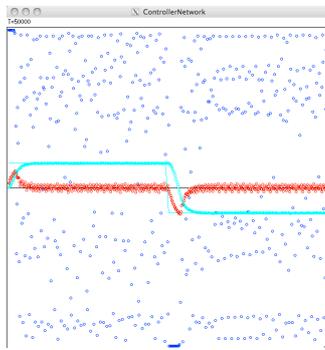
01 x 04



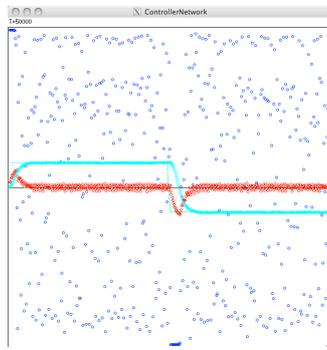
01 x 05



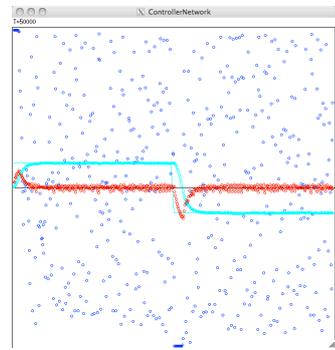
01 x 06



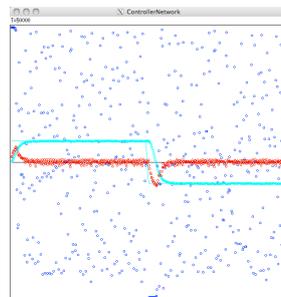
01 x 07



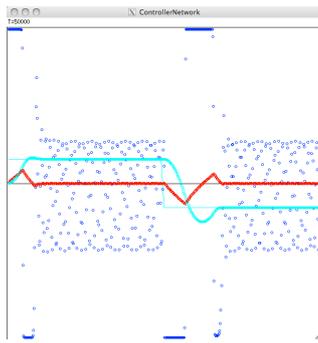
01 x 08



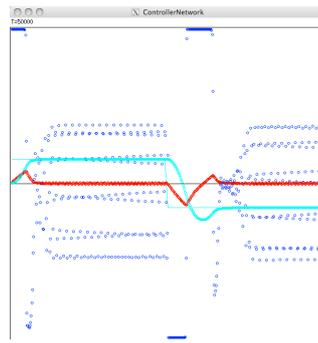
01 x 09



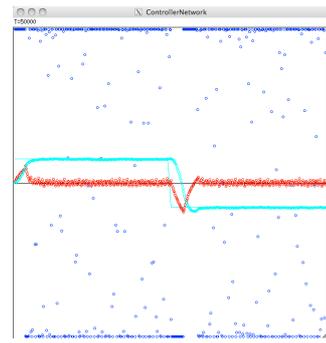
01 x 10



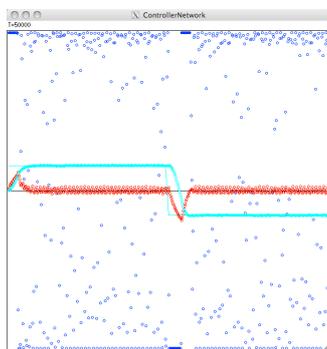
02 x 01



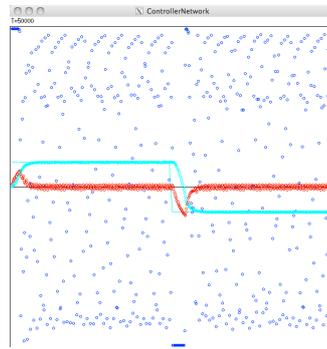
02 x 02



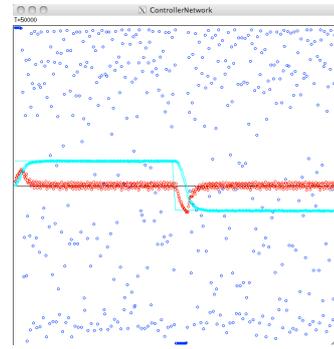
02 x 03



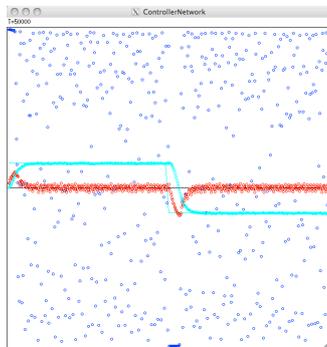
02 x 04



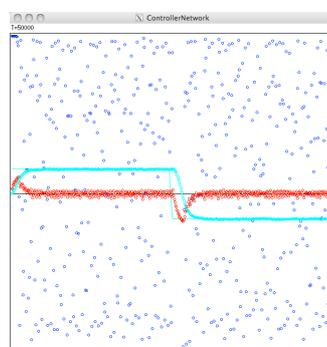
02 x 05



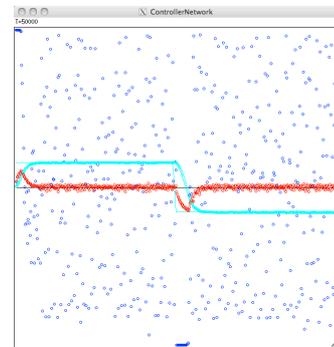
02 x 06



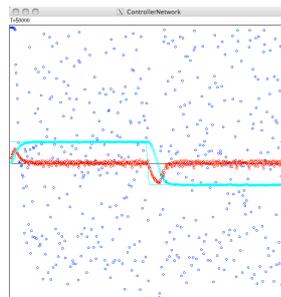
02 x 07



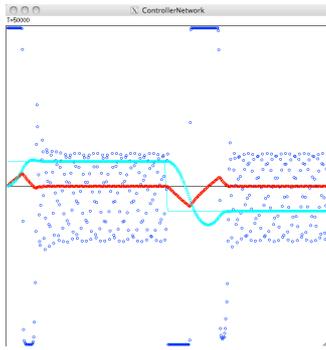
02 x 08



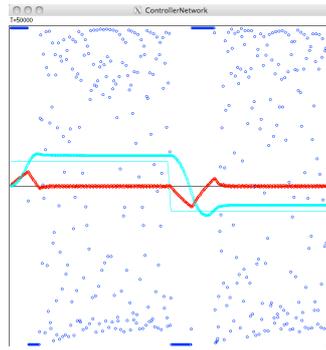
02 x 09



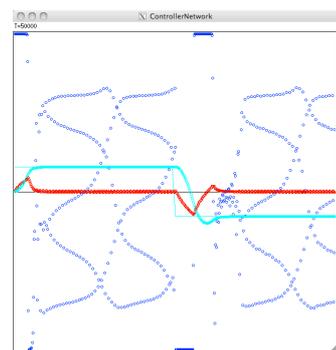
02 x 10



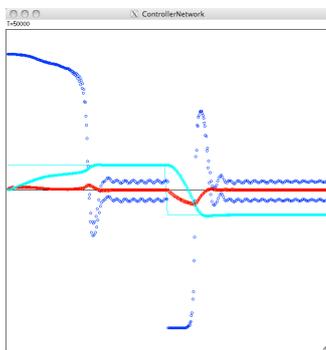
03 x 01



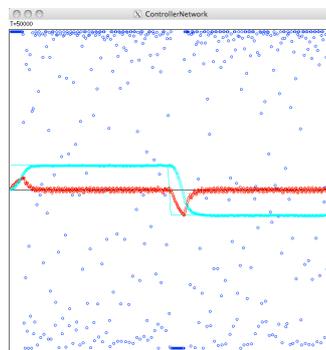
03 x 02



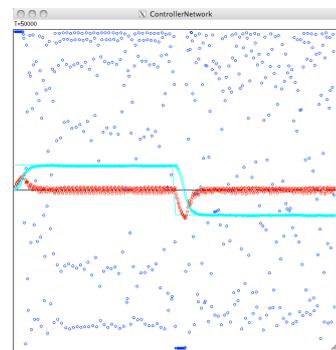
03 x 03



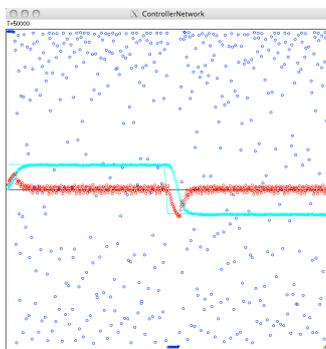
03 x 04



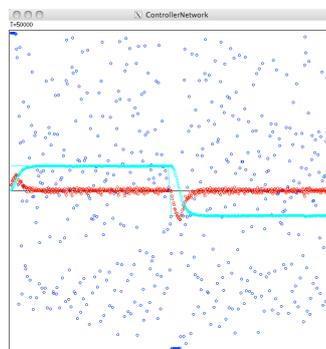
03 x 05



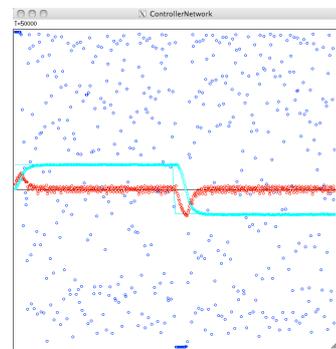
03 x 06



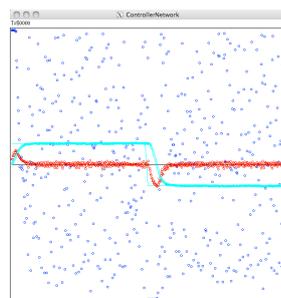
03 x 07



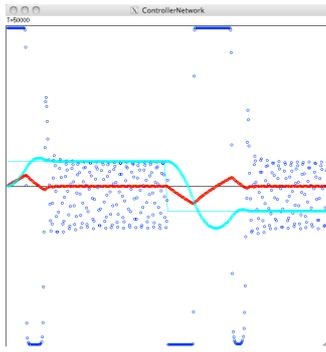
03 x 08



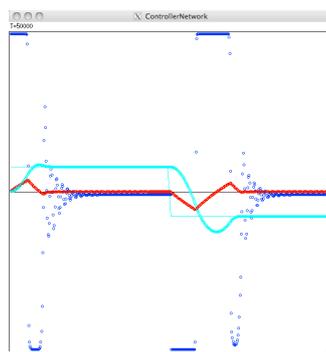
03 x 09



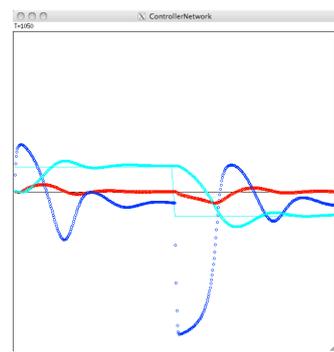
03 x 10



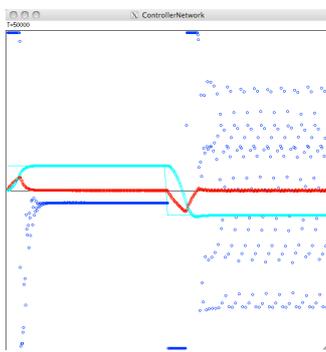
04 x 01



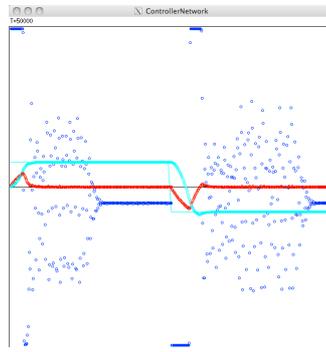
04 x 02



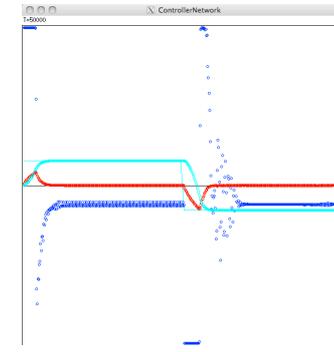
04 x 03



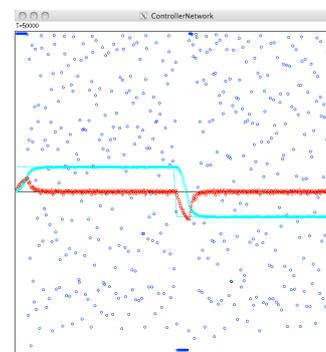
04 x 04



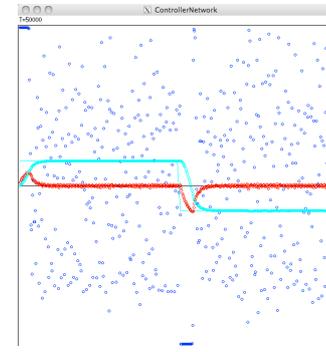
04 x 05



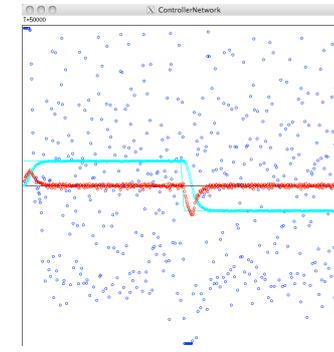
04 x 06



04 x 07

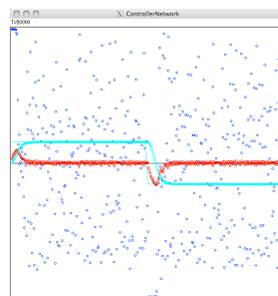


04 x 08

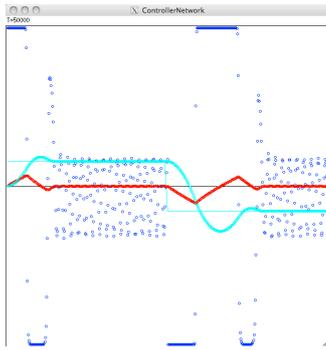


04 x 09

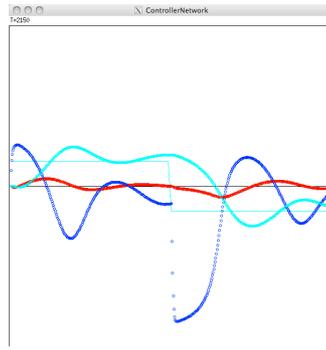
]



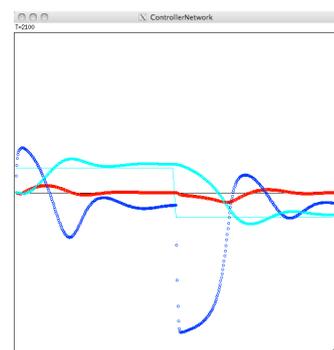
04 x 10



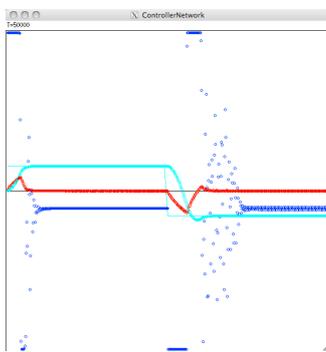
05 x 01



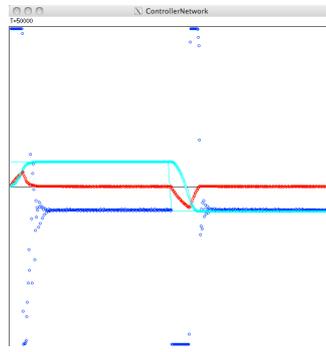
05 x 02



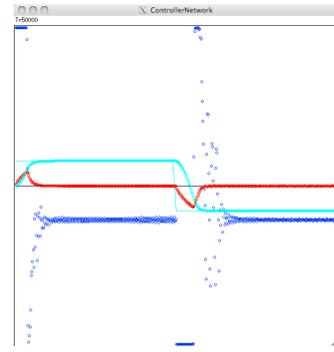
05 x 03



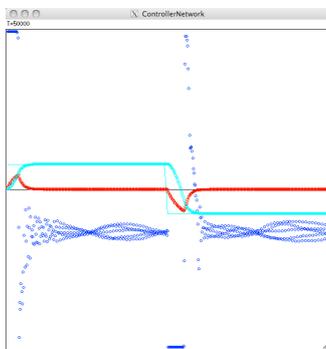
05 x 04



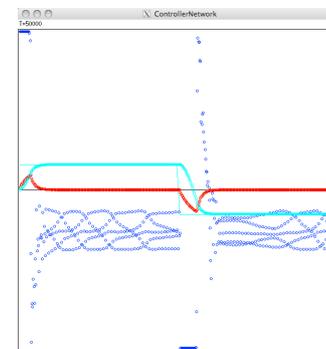
05 x 05



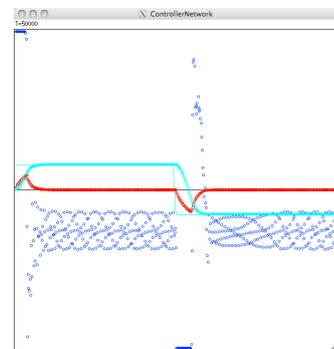
05 x 06



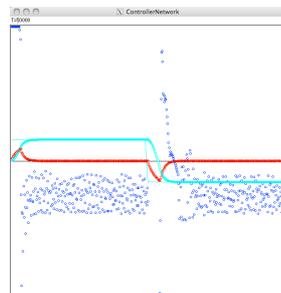
05 x 07



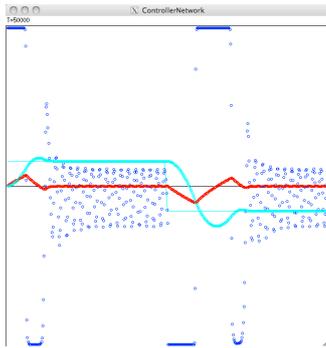
05 x 08



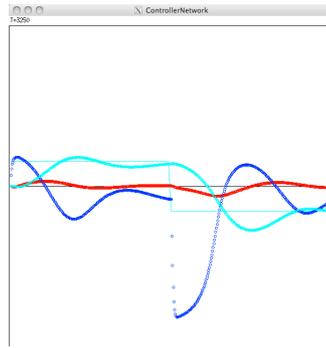
05 x 09



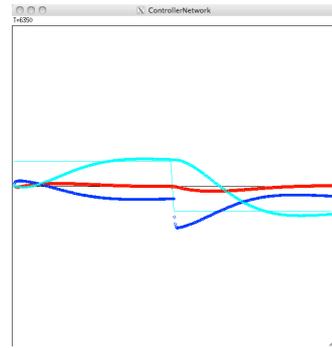
05 x 010



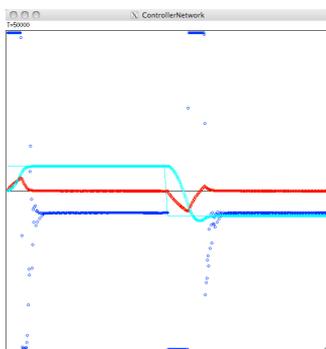
06 x 01



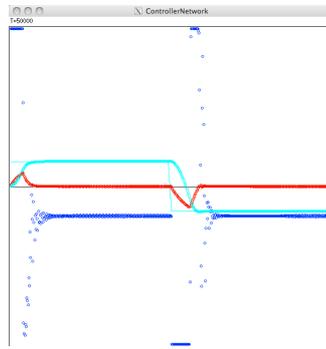
06 x 02



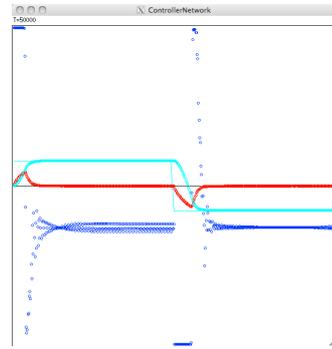
06 x 03



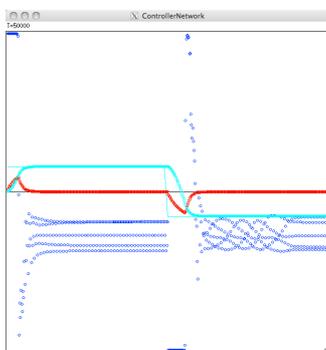
06 x 04



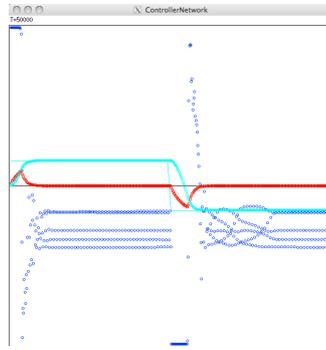
06 x 05



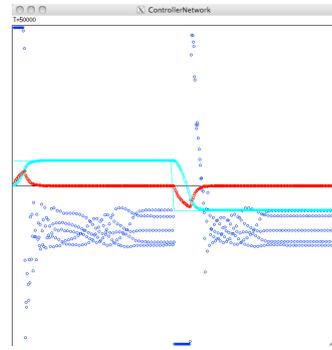
06 x 06



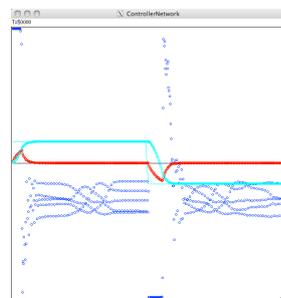
06 x 07



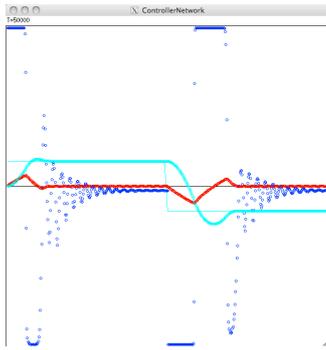
06 x 08



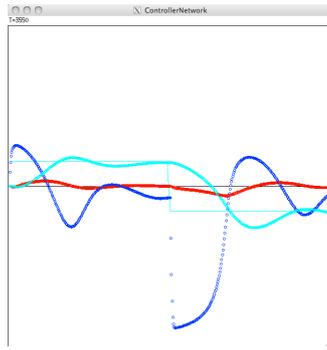
06 x 09



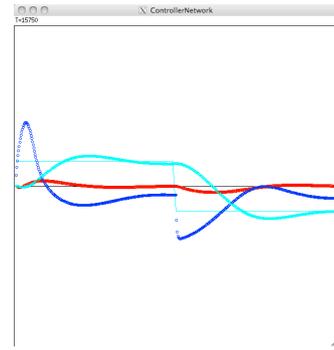
06 x 10



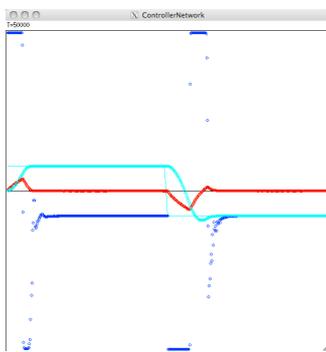
07 x 01



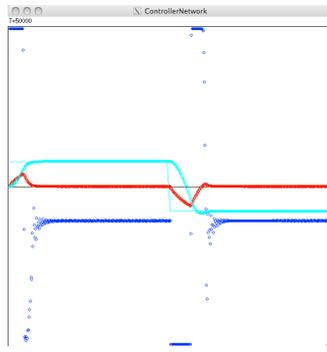
07 x 02



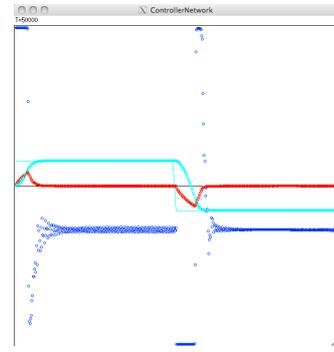
07 x 03



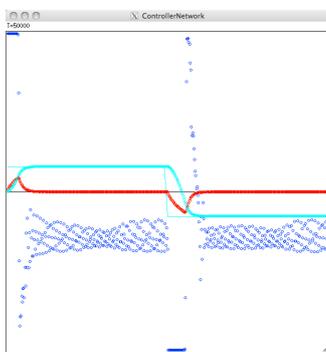
07 x 04



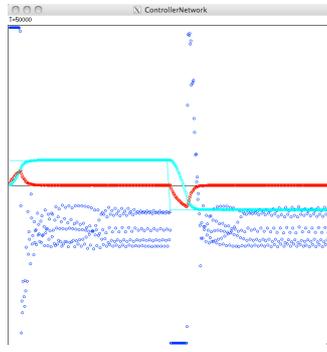
07 x 05



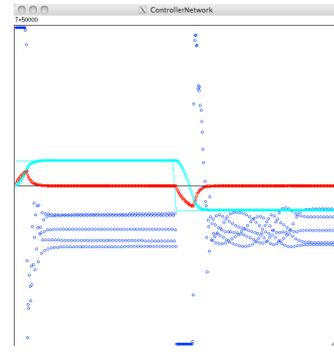
07 x 06



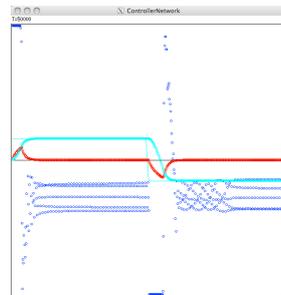
07 x 07



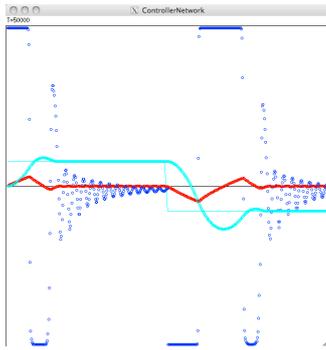
07 x 08



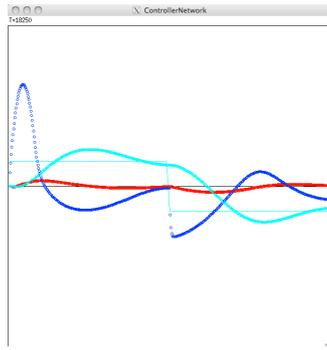
07 x 09



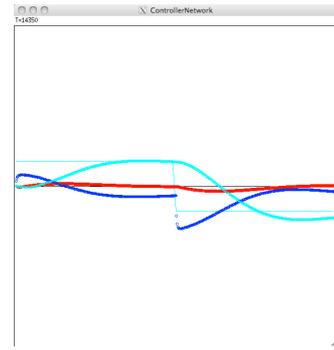
07 x 10



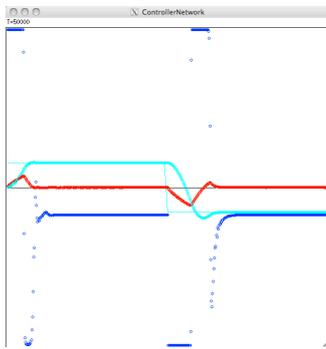
08 x 01



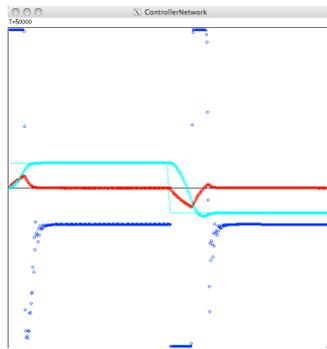
08 x 02



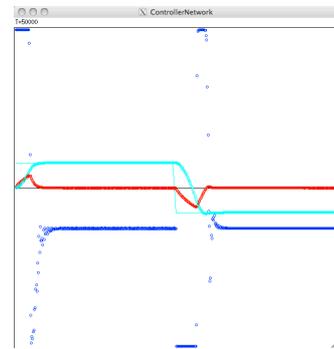
08 x 03



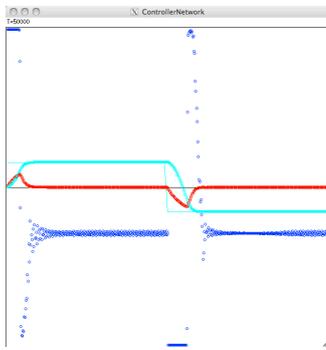
08 x 04



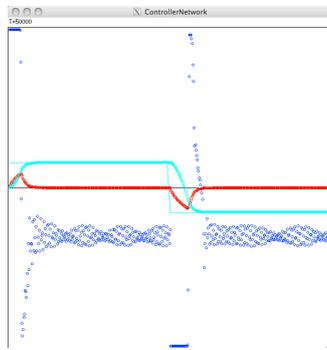
08 x 05



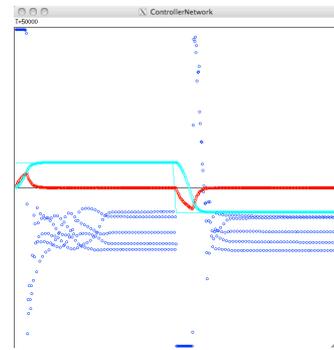
08 x 06



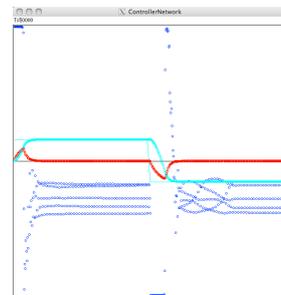
08 x 07



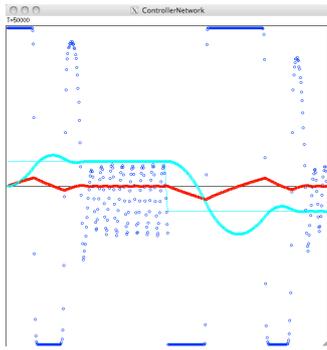
08 x 08



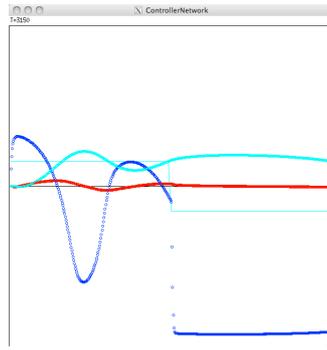
08 x 09



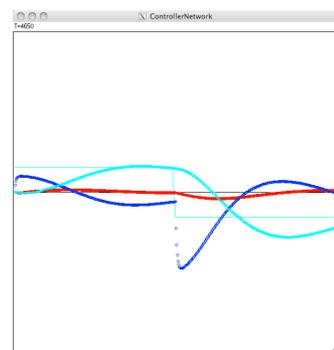
08 x 10



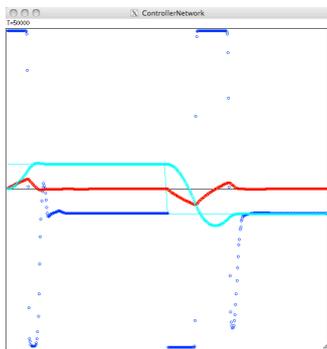
09 x 01



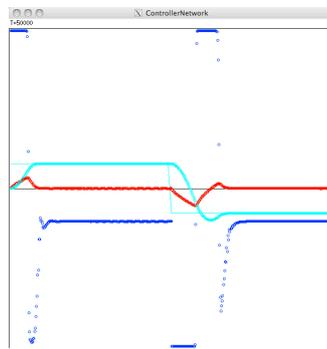
09 x 02



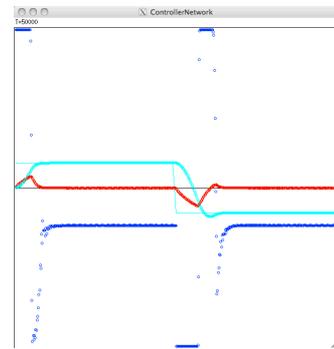
09 x 03



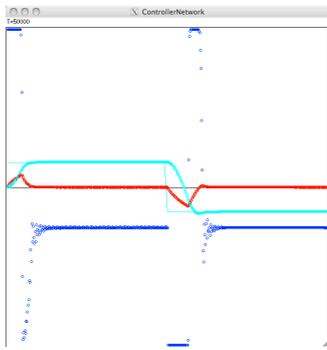
09 x 04



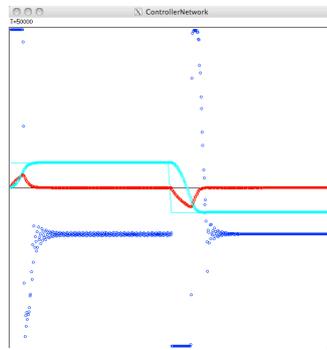
09 x 05



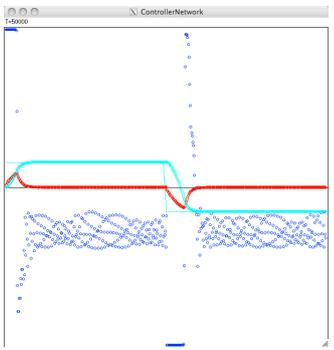
09 x 06



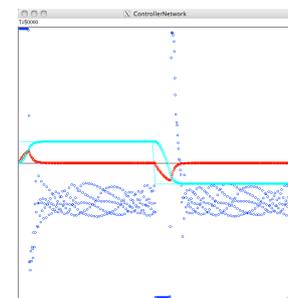
09 x 07



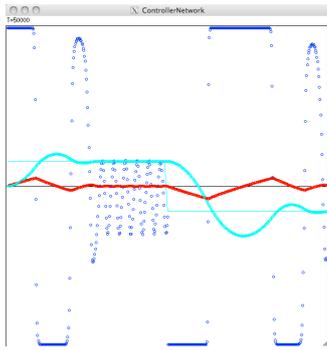
09 x 08



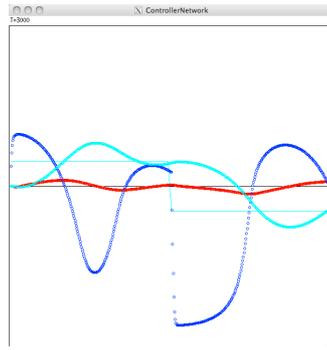
09 x 09



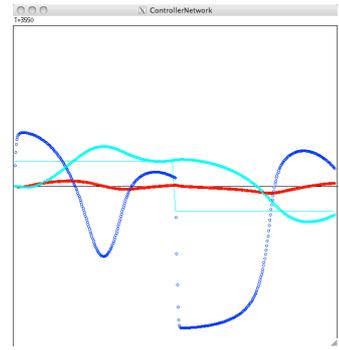
09 x 10



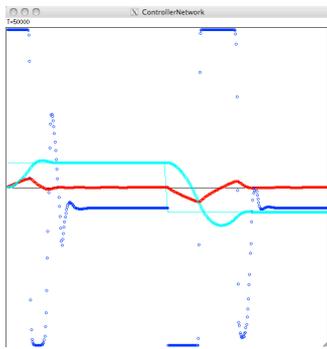
10 x 01



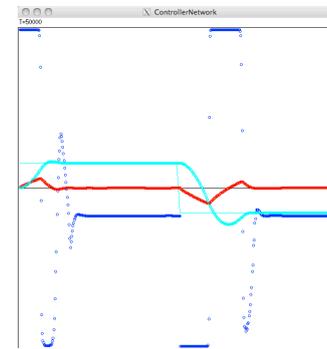
10 x 02



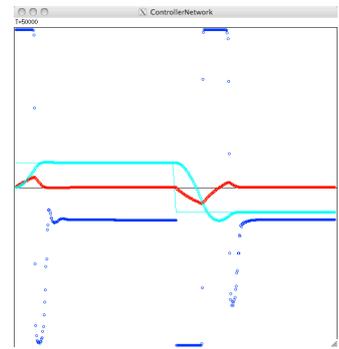
10 x 03



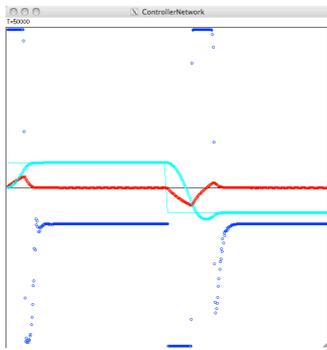
10 x 04



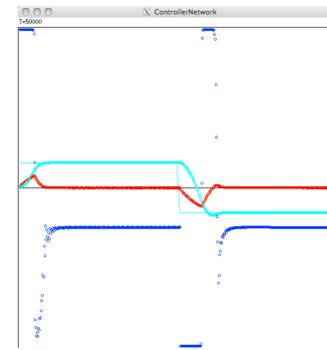
10 x 05



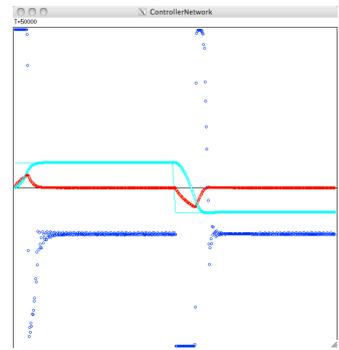
10 x 6



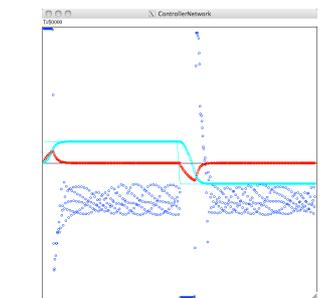
10 x 07



10 x 08



10 x 09



10 x 10

Figs. 5-21 Controller map

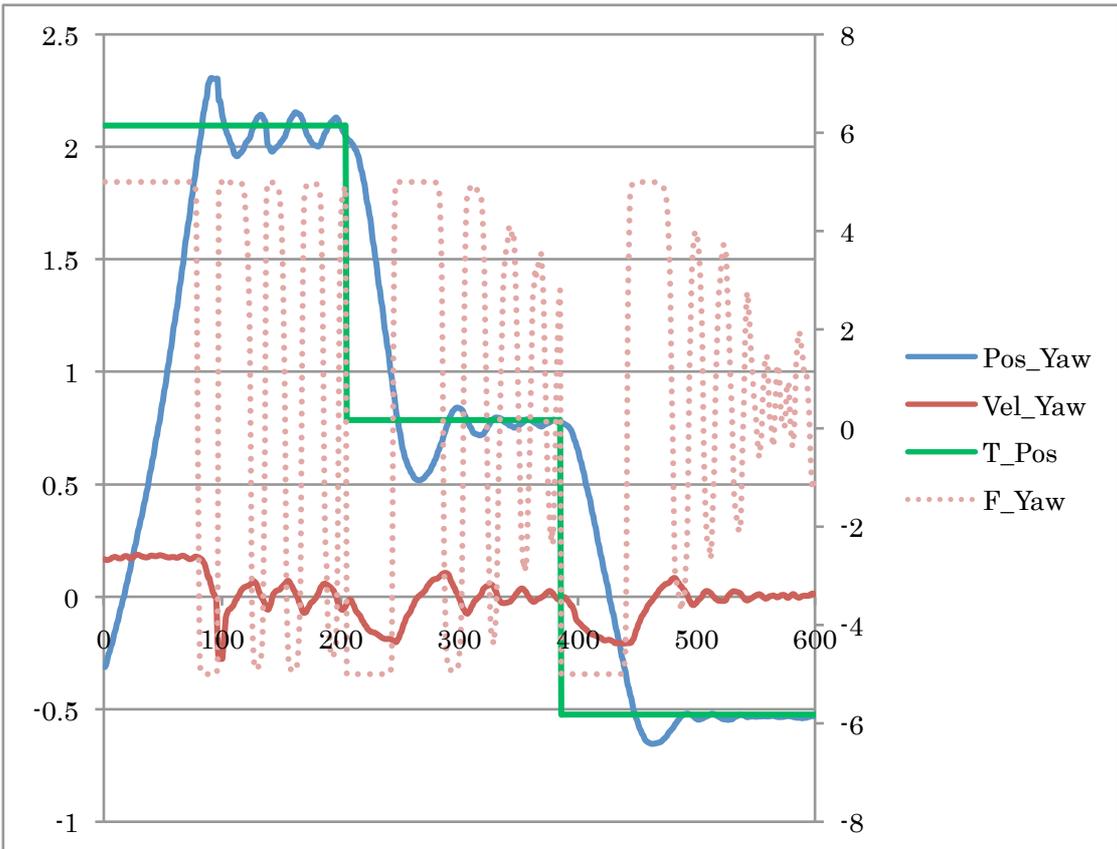


Fig. 5-22 Control Experiment of mnSOM using “Twin-Burger”

Table 5-1 Specifications of Twin-Burger

Dimensions	1.54 [m] x 0.86 [m] x 0.54 [m]
Weight	110.0 [kg]
Battery	Ni-Cd 25.2 [V] 4000 [mAh] Ni-Cd 28.8 [V] 4000 [mAh]
Durations	2.0 [h]
Sensors	Attitude Sensor ( Roll, Pitch, Yaw) Acceralation Sensor (x, y, yaw) Depth Sensor Flow sensor x 2 (Surge, Sway) Ultrasonic Range Finder x 8
Actuator	40 [W] Trusters x 5
Computer	Laptop PC

Table 5-2 Coefficient  $M$  and  $C$  for Limit Cycle Motion

M / C	10	30	50	100
10	$D_0$	$D_1$	$D_2$	$D_3$
30	$D_4$	$D_5$	$D_6$	$D_7$
50	$D_8$	$D_9$	$D_{10}$	$D_{11}$
100	$D_{12}$	$D_{13}$	$D_{14}$	$D_{15}$

## 第6章

### 考察およびまとめ

## 6.1 考察および結論

本研究は、ロボットの知能化を目的として行った。研究・開発にあたり、まず現在までの移動ロボットの開発動向について調べ、今後どのような開発を行うべきかを考察した。ロボットの歴史を紐解いていくとある今後のロボット発展を考える上で重要な言葉が浮き上がる。それらは、自律性：環境認識による行動選択の的確さ、汎用性：様々な作業に適用可能か、知能性：学習により作業精度が向上するか、身体性：環境／作業に適した形状をしているか、であり、開発コストを考慮してロボット開発が進められる。ロボットが高度な作業をする為には、ロボットは、考える、学習する、自分で状況を判断して行動を創発し、自律的に行動を行う必要がある。そこでこれらのキーワードをロボット開発に際して達成するには、ハードウェア的な知能化とソフトウェア的な知能化の双方が上手く融合する事で高度な知能を持ったロボットの開発が行う事が出来ると考えられる。

そこで、本研究では、ハードウェアの知能化を題材として、全方位移動自律サッカーロボットの開発を行った。また、ソフトウェアの知能化は、脳型情報処理アルゴリズムを用いる事でセンサを用いた環境適応型色認識アルゴリズムの開発と、西田らによって基礎開発された自律型水中ロボットの適応制御システムの開発を行った。

まず、ハードウェアの知能化として、全方位移動サッカーロボット“Musashi”の開発をアミールによって提案された“Concept of Intelligent Mechanical Design for Autonomous Mobile Robot”に基づいて開発を行った。アミールの手法では、ロボットをハードウェアから知能化していく為に、機械的設計指針として、ロボットに要求される項目（例：ロボットのタスク、必要な行動、使用する環境、環境による拘束条件など）を明らかにする事で知能化されたロボットを設計する事に成功した。また仕様を決定する際に過去に使用していたロボットの欠点から本研究で開発するロボットは、3つのコンセプトを軸として設計を行った。以下に設計指針を示す。

(a) Omni-directional Concept

(b) Modularity Concept

(c) Safety Concept

(a)を用いる事で、過去に使用したロボットの複雑な設計を変更した。以前のロボットでは、運動学的に非常に困難であった横方向や回転方向への運動を実現し、操作性や運動能力の向上につながった。また、全方位視覚の導入により、11個あったセンサを4つに減少させる事に成功した。(b)では、ロボットの各パーツをモジュール化する事でロボットのメンテナンス性や信頼性の向上、輸送の際の便性の向上等を実現した。最後に(c)によって、知能化されたロボットを社会で使用するという要求を考えるとハードウェアの知能化のアプローチとして、安全について考慮する必要がある事を提案した。移動ロボットの安全性に関する法律等は、現在存在しない。今回は、機械部品や産業用ロボットの安全基準であるISOを参考に安全な設計及び保護方策を提案した。安全に関する要求が満たされれば、様々な移動ロボットの行動範囲は確実に広がる事が出来ると考えられる。

次に脳型情報処理アルゴリズムを用いたロボットのソフトウェア上の知能化に取り組んだ。ソフトウェアの知能化のアプリケーションとして、2つのアルゴリズムを開発した。

一つ目は、照明環境適応型色認識アルゴリズムの開発を行った。これは、多くの人々に取って見える事言う事は日常的な事である。多くの人々にとって見る事なしでは、対象物を知り、その位置関係・動き等を判断する事は容易ではない。視覚のなかで、色は役に立つ重要な要素の一つである。ロボットに置いても様々な物体を同定するとき色認識手法は多く採用されている。そこで、本研究では、様々な照明環境を想定して、それらの環境にてきた色認識を行う事が出来るビジョンアルゴリズムをニューラルネットワーク等の学習アルゴリズムを用いる事で構築した。色情報は、線形和で表す事が出来ない。よって、非線形的な状態を表現できる脳型情報処理アルゴリズムを用いる事でソフトウェアの知能化を果たす事が出来るのではないかと考えた。実際に実験を通じて学習を用いた照明環境適応型色認識アルゴリズムの可能性を示す事が出来た。本研究では、自己組織化マップ(SOM)、ニューラルガス(NG)、モジュール型自己組織化マップ(mnSOM)を用いて、色認識アルゴリズムを実現したが静的(ロボットが動かない

状態)環境において、NG アルゴリズムが一番性能が良いという結果が得られた。動的環境(ロボットが時々刻々動いている状態)ではNG と mnSOM アルゴリズムがモジュール数を少なく表現しており、良い結果が得られた。また、mnSOM は環境をクラスタリングした後にニューラルネットワークによって出力を得るため環境が細かく変化する際には、mnSOM が一番有効であるのではないかと考えられる。しかし、ロボットに搭載する際にはロボットの実行速度のとのトレードオフの関係が存在する為、アルゴリズムの実行速度も考慮して判断しなければならない。以上より、脳型情報処理アルゴリズムによってセンサを用いロボットのソフトウェアとしての知能化のアプリケーションをひとつ示した。

この結果、全方位移動ロボット“Musashi”をハードウェアやソフトウェア双方から知能化して開発する事によって、以前使用していたロボットに比べ格段に性能を向上する事が出来た。実際にロボカップ中型リーグにおいて、4年間に於いて、国内大会で3度の優勝と1度の準優勝、世界大会では、最高4位という結果をおさめる事が出来た。

次に、知能化のアプリケーションの一つとして、西田によって基礎開発された mnSOM を用いた水中ロボットの運動の同定と制御システムの開発・改良を行った。動特性の異なる水中ロボットの運動時系列をリカレント型ニューラルネットワーク mnSOM を採用し、学習する事で、同時に複数の動特性を表現したリカレントニューラルネットワーク(フォワードモデル)を得るとともに、入力時系列間の関係を学習する事が出来た。同時に、フォワードモデルマップ内に入力時系列群の中間状態を表現するネットワークの獲得にも成功した。このフォワードモデルをもとに学習によって、フィードバック系のニューラルネットワークモデルを用いて、コントローラモデルを作成した。このアルゴリズムによってロボットは、過去の経験を学習し、制御を行うという知能化を行う可能性を示す事が出来た。

本研究では、知能化をテーマにロボットの開発を行ってきたが、以上に示したようにハードウェアを知能化する事で、ソフトウェアの開発への負担を軽減する事ができる。また、ハードウェアを設計する段階でロボットの安全性の向上に着目し、ロボット知能化に安全という要求を導入する事を提案した。

ソフトウェア上の知能化は脳型情報処理アルゴリズムを用いる事で、過去のデータを記憶し、学習する事で行動を制御する事に成功した。よって、知能化

のアプローチとして、脳型情報処理アルゴリズムを採用する事は有用であると  
考えられる。この2つのアプローチで開発する事で、今後のロボット開発にお  
けるキーワードとなる**自律性**、**汎用性**、**知能性**、**身体性**といったキーワード  
を上手く両方から解決する事が出来ると考えられる。

最後に、本研究は、社会的・学術的分野双方に置いて、ロボットの知能化と  
いう必要とされている分野への挑戦である。従って、本研究のような新たな知能  
化のアプローチやアプリケーションが、ロボットの開発に貢献する事が出来る  
事を期待して本論文の結びとする。

# 参考文献

## 第 1 章

- [1-1] Karel Capek 著, 千野栄一訳, “ロボット (R.U.R)”, ISBN-13-978-4003277423, 岩波書店, 2003
- [1-2] Isaac Asimov 著, 小尾美佐役, “I. ROBOT 我はロボット アシモフのロボット傑作集”, ISBN-13-978-4150114855, 早川書房, 2004
- [1-3] H.A. Ernst, “MH-1-A Computer-Operated Mechanical Hand”, *Proc. Of the AFIPS Spring Joint Computer Conference*, vol.21, pp39-51, 1962
- [1-4] 井上博允, 金出武雄, 安西祐一郎, 瀬名秀明, “ロボット創成学”, ISBN4-00-011241-4, 岩波書店, 2004
- [1-5] 経済産業省, ロボット政策研究会中間発表資料, 2006  
<http://www.meti.go.jp/policy/robotto/oldpress.html>
- [1-6] 久保田孝, “火星探査機”Mars Pathfinder”, 日本ロボット学会誌, Vol.15, No.7, pp.986-992, 1997
- [1-7] 火星探査ロボット sojourne: <http://mars.jpl.nasa.gov/MPF/rover/sojourner.html>
- [1-8] 火星探査ロボット opportunity: <http://marsrover.nasa.gov/home/index.html>
- [1-9] 浦環, “自律型海中ロボット r2D4 の製作と佐渡沖および黒島海丘海底観測”, 日本ロボット学会誌, Vol.22, No.6, pp.709-713, 2004
- [1-10] Ura T., Nagahashi K., Asada A., Okamura K., Tamaki K., Sakamaki T., Iizasa K., “Dive into Myojin-sho Underwater Caldera”, *Proc. of OCEANS'06*, 2006
- [1-11] Reber, A.S., *dictionary of psychology*, p.379, London, Penguin Books, 1995
- [1-12] *the Journal of Educational Psychology*, Vol.12, pp.123-137, pp.195-216, 121
- [1-13] Webb, B., “Perception in real and artificial insects: A robotic investigation of cricket phonotaxis”, Unpublished doctoral dissertation, University of Edinburgh, Scotland
- [1-14] Webb, B., “Robotic experiments in cricket phonotaxis. In D. Cliff, P. Husbands, J.-A. Meyer, and S.W. Wilson (Eds.), *From animals to animals: Proceedings of the Third International Conference on Simulation of Adaptive Behavior.* , pp.45-54, Cambridge, MA: MIT Press , a Bradford book
- [1-15] Lambrions, D., Maris, M., Kobayashi, H., Labhart, T., Pfeifer, R., and Wehner, R., “2 An autonomous agent navigating with a polarized light compass”, *Adaptive Behavior*,

Vol.6, pp.175-206, 1997 また, Ferrell ら

[1-16] Ferrell, C. , Maes, P., Mataric, M., Meyer, J.-A., Pollach, J. and Wilson, S.W., “Orientaton behavior using registered topographic maps.”, From Animals to Animats, Vol.4, Cambridge, Ma: MIT Press, pp.94-104, 1996

[1-17] Cruse, H., “Coordination of leg movement in walking animals. “, From animals to animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior, pp.105-119, Cambridge, MA: MIT Press, 1991

[1-18] Philip Johnson-Laird, “The Computer and the Mind: An introduction to Cognitive Science”, 448 pages, Harvard University Press, ISBN-13:978-0674156166, 1998

[1-19] R. Pfeifer, C.Scheier 著, 石黒章夫, 小林宏, 細田耕監訳, “知の創成 –身体性認知科学への招待–”, 695 pages, ISBN4-320-1203-2-9, 共立出版, 2001

## 第 2 章

[2-1] 佐藤雅紀, 石井和男, “不整地走行を目的とした車輪型移動ロボットのニューロコントローラによる制御”, 第 2 2 回日本ロボット学会学術講演会予稿集, 2I14(2 pages), 2004

[2-2] 佐藤雅紀, 石井和男, “不整地走行を目的とした車輪型移動ロボットのコントロールシステムの設計と実装”, IIP2005 情報・知能・精密機器部門講演会講演論文集, pp.281-286, 2005

[2-3] 佐藤雅紀, 石井和男, “不整地走行を目的とした車輪型移動ロボットの制御系設計”, ロボティクス・メカトロニクス講演会'05 講演会論文集, 2A1-S-015(1)-(4), 2005

[2-4] Masanori, Sato, Atushi Kanda, Kazuo Ishii, “A Switching Controller System Based on Environment Recognition”, Proc. Of the International Conference on Intelligent Unmanned System, pp.250-257, 2007

[2-5] Masanori Sato, Atushi Kanda, Kazuo Ishii, “ A Switching Controller System for a Wheeled Mobile Robot”, Journal of Bionic Engineering, Vol.4, No.4, pp.281-289, 2007

[2-6] K. Ishii, S.Nishida, T.Ura, “ A Self-Organizing Map Based Navigation System for an Underwater Vehicle”, Proc. Of ICRA'04, pp.4466-4471, 2004

[2-7] 石川眞澄, “ニューラルネットワークの学習の数理”,九州工業大学大学院

- 生命体工学研究科脳情報専攻高次脳情報演習配布資料, pp.4~5
- [2-8] 石川眞澄, 忘却を用いたコネクショニストモデルの構造学習アルゴリズム, 人工知能学会誌, 5 巻, 5 号, pp.595-603 (1990)
- [2-9] Yasui,S. “Convergence suppression and divergence facilitation: Minimum and joint use of hidden units by multiple outputs.”, *Neural Networks*. 10, 2: 353-367, 1997
- [2-10] 古川徹生, “脳型パターン認識 ～第7回: 多層パーセプトロンと誤差逆伝播学習”, 九州工業大学生命体工学研究科脳型パターン認識配布資料, pp.2~5
- [2-11] T.Kohonen, “Self Organizing Maps”, pringer-Verlag, 1995
- [2-12] T.Kohonen, “self-organized formation of topologically correct feature maps”, *Biological Cybernetics*, Vol.43, pp.59~69, 1982
- [2-13] Samuel Kaski, Jari Kangas, T.Kohonen, “Bibliography of Self-Organizing Map(SOM) Papers: 1981-1997”, *Neural Computing Surveys* 1, pp.102~350, 1998
- [2-14] 徳高平蔵, 藤村喜久朗, 山川烈監修: 自己組織化マップ応用事例集, SOMによる可視化情報処理, 海文堂, 2002
- [2-15] 大谷誠, “modular network SOM(mnSOM)の正規直交関数展開による理論・シミュレーション両面からの動作原理”, COE マルチタレント英才教育発表論文, pp.4~5
- [2-16] Simon Haykin 著, “Neural Networks ~A Comprehensive Foundation~”, Prentice Hall International Inc., ISBN 0-13-908385-5, Chapter 9, Self Organizing Map, pp.443-48
- [2-17] T.M.Martinetz, K.J.Schulten, “A “neural-gas” network learns topologies”, In T.Kohonen, K. Makisara, O.Simula, and J.Kangas, editors, *Artificial Neural Networks*, pages 397-402, North-Holland, Amsterdam, 1991
- [2-18] T.M.Martinetz and K.J.Schulten, “Topology representing networks”, *Proc. Neural Networks*, Vol.7(4.), :-522, 1994
- [2-19] Tetsuo Furukawa, Kazuhiro Tokunaga, Kenji Morishita, Syozo Yasui, “Modular Network SOM(mnSOM): From Vector Space to Function Space”, *International Joint Conference on Neural Networks(IJCNN)*, 2005

### 第 3 章

- [3-1] Amir A.F. Nassiraei, “Concept of Intelligent Mechanical Design for Autonomous Mobile Robots”, *Journal of Bionic Engineering*, Vol.4, no.4. pp.281-289, 2007
- [3-2] Amir A.F. Nassiraei and K. Ishii, “How Does “Intelligent Mechanical Design

Concept” Help Us to Enhance Robot’s Function?”, Intelligent Unmanned System, Theory and Applications, Studies in Computational Intelligence, Vol.92/2009, pp.155-178,2009

[3-3] Amir A.F. Nassiraei, “Concept of Intelligent Mechanical Design for Autonomous Mobile Robots”, Kyushu Institute of Technology Ph.D dissertation, 2007

[3-4] H.Kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawa, and H. Matsubara, “robocup: A challenge problem of ai”, *AI magazine*, Vol.18, No.1, pp.73-85, 1997

[3-5] 北野宏明, 浅田稔, “「ワールドカップ」ロボットの挑戦”, *日経サイエンス*, Vol.28, pp.74-82, 1998

[3-6] H. Kitano, ed., “*RoboCup-97: Robot Soccer World Cup I*”, Springer, Lecture Note in Artificial Intelligence 1395, 1998

[3-7] M.Asada and H.Kitano, eds. “*RoboCup-98:Robot Soccer World Cup II*”, Springer, Lecture note in Artificial Intelligence 1604, 1999

[3-8]M.Veloso, E.Pagello, and H.Kitano, eds. “*RoboCup-99: Robot Soccer World Cup III*”, Springer Lecture note in Artificial Intelligence 1856, 2000

[3-9] P.stone, T.Balch, and G. Kraetzschmar, eds. “ *RoboCup- 2000: Robot Soccer World Cup IV*”, Springer, Lecture Note in Artificial Intelligence 2019, 2001

[3-10]A.Birk, S.Coradeschi, and S.Tadokoro, eds. “ *RoboCup- 2001: Robot Soccer World Cup V*”, Springer, Lecture Note in Artificial Intelligence 2377, 2002

[3-11] G.Kaminka, P.U. Lima, and R.Rojas, eds. “ *RoboCup- 2002: Robot Soccer World Cup VI*”, Springer, Lecture Note in Artificial Intelligence 2752, 2003

[3-12] B.Browning, A.Bonarini, D.Polani and K.Yoshida, eds. “ *RoboCup- 2003: Robot Soccer World Cup VII*”, Springer, Lecture Note in Artificial Intelligence 3020, 2004

[3-13] C.Sammut, J.Santos-Victor, D.Nardi, M.Riedmiller, eds. “ *RoboCup- 2004: Robot Soccer World Cup VIII*”, Springer, Lecture Note in Artificial Intelligence 3276, 2005

[3-14] A.Bredenfeld, A.Jacoff, I. Noda and Y.Takahashi, eds. “ *RoboCup- 2005: Robot Soccer World Cup IX*”, Springer, Lecture Note in Artificial Intelligence 4020, 2006

[3-15] G.Lakemeyer, E.Sklar, D.G.Sorrenti and T.Takahashi, eds., “*RoboCup- 2006: Robot Soccer World Cup X*”, Springer, Lecture Note in Artificial Intelligence 4434, 2007

[3-16] U.Visser, F.Ribeiro, T.Ohashi, F.Dellaert, eds. “ *RoboCup- 2007: Robot Soccer*

- World Cup XI*’, Springer, Lecture Note in Artificial Intelligence 5001, 2008
- [3-17] L.Locchi, H.Matsubara, A.Weitzenfeld, C.Zhou, eds. “ *RoboCup- 2008: Robot Soccer World Cup XII*”, Springer, Lecture Note in Artificial Intelligence 5399, 2009
- [3-18] M.Asada, E.Pagello, T.Balch, F.Ribero, Y.Takemura, et. al., “RoboCup Middle Size Robot League Rules and Regulations for 2009”, <http://www.er.ams.eng.osaka-u.ac.jp/robocup-mid/index.cgi?page=Rules+and+Regulations>, Nov. 14, 2008
- [3-19] NPO ロボカップ日本委員会監修, 中村恭之, 高橋泰岳編著, “中型ロボットの基礎技術 -対戦のための協調行動に向けて-”, 共立出版社, ISBN4-320-12153-8, 2005
- [3-20] Amir A.F. Nassiraei, “Concept of Intelligent Mechanical Design for Autonomous Mobile Robots”, *Journal of Bionic Engineering*, Vol.4, no.4. pp.281-289, 2007
- [3-21] Amir A.F. Nassiraei and K. Ishii, “How Does “Intelligent Mechanical Design Concept” Help Us to Enhance Robot’s Function?”, *Intelligent Unmanned System, Theory and Applications, Studies in Computational Intelligence*, Vol.92/2009, pp.155-178,2009
- [3-22] Amir A.F. Nassiraei, “Concept of Intelligent Mechanical Design for Autonomous Mobile Robots”, Kyushu Institute of Technology Ph.D dissertation, 2007
- [3-23] A. Bredenfeld et. al, “GMD-Musashi team description”, *RoboCup 2003, Italy, Proc.*
- [3-24] T. Buchhei, et. al., “Team Description Paper 2005 CoPS Stuttgart”, *RoboCup 2005 Osaka, CD-ROM Proc. of RoboCup 2005*
- [3-25] ISO 12100-2: Safety of machinery - Basic concepts, general principles for design - Part 2: Technical principles, 2003
- [3-26] ISO10218-1: Robots for industrial environments -Safety requirements - Part 1: Robot, 2006
- [3-27] ISO 14121-1: Safety of machinery - Risk assessment - Part 1: Principles, 2007
- [3-28] ISO/TR 14121-2: Safety of machinery - Risk assessment - Part 2: Practical guidance and examples of methods, 2007
- [3-29] A.A.Nassiraei, S.Masakado, et. al, “Development of an Artistic Robot “JumpingJoe””, *Proc. Of IEEE IROS 2006*, pp.1720-1725, 2006

[3-30] A.A.F. Nassiraei, S.Masakado, et. al, “Development of an artistic robot “Jumping Joe””, *Brain-Inspired IT II*, pp.281-284, Elsevier, 2006

[3-31] A.AF. Nassiraei, K.Ishii, et. al, “Realization of Rapid Movement for Legged Entertainment Robots Using Two New Actuators, the Inertia Actuator and the Cam Charger”, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol.11, No.8, pp.979-988,2007

[3-32]小川優, et. Al, “サービスロボットにおけるリスクアセスメントと保護方策に関する試み”, ロボティクスメカトロニクス学術講演会 2009, 福岡

## 第 4 章

[4-1] 飯山将晃, 青木啓史, 亀山能成, 美濃導彦, “未知光源環境下での物体形状と反射性のボクセル独立な再構成法”, 情報処理学会誌, Vol.42, No.12, pp.3185-3193,20001

[4-2] Yoichi Sato, Mark D.Wheeler and Katsushi Ikeuchi, “Object Shape and Reflectance Modeling from Observation”, Proc. SIG-GRAPH 97 Conference, pp.379-387,1997

[4-3] 川上玲, タン ロッビタントウイ, 池内克史, “光源環境の変換を利用した物体の色推定”, 画像の認識理解シンポジウム(MIRU2004), pp.II-169-174,2004

[4-4]松山隆司, 久野義徳, 井宮淳, “コンピュータビジョン技術評価と将来の展望”, Chapter 5, 新技術コミュニケーションズ, 19984

[33] Shree K. Nayar, Katsushi Ikeuchi and Takeo Kanade, “Surface Reflection:Physical and Geometrical Perspectives”, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.13, No.7, pp.611-634, 1991

[4-5] 市川修司, 出村公成, “照明変化にロバストな物体識別”, 日本ロボット学会第 29 回学術講演会, CD-Proc., 2I2-01, 2009, 横浜

[4-6] Gegenfurtner, Rieger, “Sensory and cognitive contributions of color to the recognition of natural scenes”, *Current Biol.*10, pp.805~808,2000

[4-7] 三上章允, “脳の世界 ～感覚のしくみ～ 色は脳でみる”, Section of Brain Research, Dept. of Behavioral and Brain Sciences, Primate Research Institute, Kyoto University homepage, <http://www.pri.kyoto-u.ac.jp/brain/brain/25/index-25.html>

[4-8] F.デルコミン著,小倉明彦・富永恵子訳, “ニューロンの生物学”, 第 11 章視

覚系, pp.235~266, 南江堂, ISBN4-524-22431-9

[4-9] 畦浦和人, Ivan Godler, “前方位カメラを用いたロボットサッカーにおける色抽出”, 日本ロボット学会学術講演会予稿集, CD-ROM, 1B25, 2006, 岡山

[4-10] Y.Takemura, Y.Sato, K.Azeura, I.Godler, K.Ishii, “SOM based on Color Constancy Algorithm for RoboCup robot”, SCIS & ISIS 2008, CD-Proc. SA-F3-5, 2008, Nagoya

[4-11] 武村泰範, 石井和男, “学習により照明環境を判別する色認識システムの性能評価と検討”, 日本ロボット学会第29回学術講演会, CD-Proc., 1R2-06, 2009

[4-12] Y.Takemura, K.Ishii, “Development of the Color Constancy Vision Algorithms using Bio-inspired Information Processing”, IEEE International Conference on Systems, Man and cybernetics 2009 (SMC 2009), Proc, P720.pdf, 2009, San Antonio TX, USA

[4-13] Y.Takemura, K.Ishii, “Evaluation of Color Constancy Vision Algorithm for Mobile robot” IEEE International Conference on ICONIP 2009, Bangkok, Tai land

## 第 5 章

[5-1] 山崎徹夫, “銅が危ない！-深海底鉱物資源開発の必要性と可能性-”, KANRIN 日本船舶海洋工学会誌, 第6号, pp.64-70,2006

[5-2]高井研, “地球最後のフロンティア近くに存在する微生物たち”, 海と地球の情報誌 Blue Earth 通巻第66号, pp8-11,2003

[5-3]浦環, 高川真一:海中ロボット総覧, 成山堂書店, 1994

[5-4]T.Ura, “Free Swimming Vehicle PTEROA for Deep Sea Survey”, Proc. Of ROV'89, pp.263-268,1989

[5-5] T. Fujii, et. Al, “Development of a versatile test-bed “Twin-Burger” toward realization of intelligent behaviors of autonomous underwater vehicles, “ Proc. Of OCEANS'93, vol.1, pp.I186-I191, 1993

[5-6] T,Ura, “Development of autonomous underwater vehicles in Japan”, Advanced Robotics, Vol.6, No.1, pp.3-15, 2002

[5-7] S.Ishibashi, T.Aoki, et al, “An Ocean Going Autonomous Underwater Vehicle “URASHIMA” equipped with a Fuel Cell”, Proc. Of UT'04, pp209-213, 2004

[5-8] 熊谷道夫, 石川加奈子, “自律型潜水ロボット「淡探」を用いた琵琶湖研究の高度化と展望”, 日本造船学科師, no.883,pp52-56,2005

- [5-9] 大畑智海, “沿岸域における観測を目的とした小型水中ロボットのシステムの開発”, 2006年度九州工業大学博士論文
- [5-10]S.Lerner, D.Fornari, B.Walden:“The Alvin Frame-Grabber System” DRAFT TECHNICAL REPORT,Wood Hole Oceanographic Institution,2003
- [5-11]A.Bennett,J.Leonard,J.Bellingham : “Bottom Following for Survey-Class Autonomous Underwater Vehicles”,Proc.UUST,pp.327-336,1995
- [5-12] R.Stoley,A.Roup : “Development of the REMUS 600 Autonomous Underwater Vehicle”,CD-ROM,Proc.Oceans'05,4pages,2005
- [5-13]WHOI,MarineOperations,Jason/Medea:<http://www.whoi.edu/marops/vehicles/jason>
- [5-14]International Submarine Engineering Ltd. : <http://www.ise.bc.ca/index.html>
- [5-15]PAN-MOOK LEE, BONG-HUAN JUN, et al.:“Navigation and Control System of Deep-sea Unmanned Underwater Vehicle‘HEMIRE’”
- [5-16]藤岡換太郎: “潜水調査船「しんかい 6500」とその成果”,日本造船学会誌,no.883,pp3-7,2005
- [5-17]渡邊正之,橋本菊雄ら: “10,000m 級無人探査機「かいこう」の活躍と亡失”,日本造船学会誌,no.883,pp22-25,2005
- [5-18]Tamaki Ura: "An Autonomous Vehicle R1 for One-Day Investigation over Mid-Ocean Ridge", InterRidge News Letter No.4, (1992)
- [5-19]浦環: “自律型海中ロボット r2D4 の製作と佐渡沖および黒島海丘海底観測”,日本ロボット学会誌,Vol.22,no6,pp.19-23,2004
- [5-20]小原敬史: “アールワン・ロボットから r2D4 へ”,日本造船学会誌,no.883,pp42-47,2005
- [5-21]白崎: “海底ケーブル観測用 AUV の研究から商用化まで”,日本造船学会誌,no.883,pp48-51,2005
- [5-22]H. Kondo, T. Ura,"Underwater Structure Observation by the AUV with Laser Pointing Device", Proc. UT02, pp.178-183, (2002)
- [5-23]T.Maki,H.Kondo,T.Ura,T.Sakamaki:“Navigation of an Autonomous Underwater Vehicle for Photo Mosaicing of Shallow Vent Areas”,CD-ROM,Proc.of OceansASIA'06, 7 pages,2006
- [5-24]巻俊宏,近藤逸人,浦環,坂巻隆: “自律型水中ロボット「トライドッグ 1 号」による防波堤の観測”,CD-ROM, Proc.ROBOMECH2005, 2P1-S-013 4 pages,2005

- [5-25] S.Ohata,Y.Eruguchi,K.Ishii:“AquaBox : A Small Underwater Robot System for Coastal Observation”,Workshop for Asian and Pacific Universities' Underwater Roboticians(APuuRobo 2006)
- [5-26]大畑智海, 江里口優, 石井和男, “AquaBox:沿岸域における観測を目的とした小型水中ロボットシステム -AUVSI & ONR’ s 9 th International Autonomous Underwater Vehicle Competition 報告 - ” 計測自動制御学会 (SI2006) CD-ROM,Proc.Of SI2006, 2 pages
- [5-27] S.Ohata,Y.Eriguchi, K.Ishii, AquaBox Series: Handy Underwater Robots for Shallow Water, 9th International AUV Competition Journal Paper, pp. KIT-1 to 6, 2006, 査読有り
- [5-28] 浦環, 高川慎一, "海中ロボット", 成山堂書店, 1997.
- [5-29] 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 グライディング航行の研究)", 日本造船学会論文集, Vol.162, pp.117-124, 1987.
- [5-30] 前田久明他, "無索無人潜水艇に働く線形流体力及び操縦応答に関する研究", 日本造船学会論文集, Vol.164, pp.211-220, 1988.
- [5-31] 大楠丹, 柏木正, 小寺山亘, "Towed Vehicle の動力学に関する基礎的研究", 日本造船学会論文集, Vol.162, pp.99-109, 1987.
- [5-32] J.M. Zurada, R.J. Marks II, C.J. Roinson, "Computational Intelligence, Imitating Life", IEEE Press, 1994.
- [5-33] M. Palaniswami, Y. Attikiouzel, R.J. Marks II, David B. Fogel and T. Fukuda, "Computational Intelligence, A Dynamic System Perspective", IEEE Press, 1995.
- [5-34] T. Ura, T. Fujii, Y. Nose and Y. Kuroda, "Self-Organizing Control System for Underwater Vehicles", Proc. of OCEANS'90, pp.76-81, 1990.
- [5-35] 藤井輝夫, 浦環, 黒田洋司, 能勢義昭, "自己生成型ニューラルネットコントローラシステムの開発と潜水機の運動制御への適用(その2:フォワードモデルの改良と無索テストベッドによる実験)", 日本造船学会論文集, Vol.169, pp.477-486, 1991.
- [5-36] T. Fujii and T. Ura, "SONCS: Self-Organizing Neural-Net-Controller System for Autonomous Underwater Robots", Proc. of IEEE/INNS IJCNN'91, pp.1973-1982, 1991.
- [5-37] K. Ishii, T. Fujii and T. Ura, "An on-line adaptation method in a neural network based control system for AUVs", IEEE Journal of Oceanic Engineering, Vol.20, No.3, pp.221-228, 1995.

- [5-38]浦環, 須藤拓, "自己訓練による海中ロボットの定高度航行", 日本造船学会論文集, Vol.171, pp.581-586, 1992.
- [5-39] 浦環, 須藤拓, "自己訓練による海中ロボットの定高度航行(その2:フォワードモデルの改良)", 日本造船学会論文集, Vol.174, pp.917-924, 1993.
- [5-40]J. Yuh, "Leaning Control for Underwater Robotic Vehicles", IEEE Control System Magazine, Vol.15, No.2, pp.39-46, 1994.
- [5-41]J. Yuh, R. Lakshmi, "An Intelligent Control System for Remotely Operated Vehicles", IEEE J. of Oceanic Engineering, Vol.18, No.1, pp.55-62, 1993.
- [5-42]J. S. Wang, C. S. G. Lee, J. Yuh, "An On-line Self-Organizing Neuro-Fuzzy Control for Autonomous Underwater Vehicles", Proc. of ICRA'02, pp.1095-1100, 2002.
- [5-43]J. Guo, F. C. Chiu, C. C. Wang, "Adaptive Control of an Autonomous Underwater Vehicle Testbed Using Neural Networks", Proc. of Oceans'95, pp.1033 - 1039, 1995.
- [5-44]J. H. Li, P. M. Lee, S. J. Lee, "Neural Net Based Nonlinear Adaptive Control for Autonomous Underwater Vehicles", Proc. of ICRA'02, pp.1075 - 1080, 2002.
- [5-45]P. J. Kodogiannis, G. Lisboa, J. Lucas, "Neural Network Modeling and Control for Underwater Vehicles", Artificial Intelligence in Engineering, Vol.1, pp.203-212, 1996.
- [5-46]R.S. Burns, "The Use of Artificial Neural Networks for the Intelligent Optimal Control of Surface Ships", IEEE Journal of Oceanic Engineering, Vol.20, No.1, pp.65-72, 1995.
- [5-47]小川原陽一, 岩本才次, 吉村学, "風外乱補償機能を付加した船舶操縦運動の学習型フィードフォワード制御方式の基礎的検討", 日本造船学会誌論文集, Vol.178, pp.321-328, 1995.
- [5-48]I. Yamamoto, Y. Terada, T. Nagamatu and Y. Imaizumi, "Propulsion System with Flexible/Rigid Oscillating Fin", IEEE Journal of Oceanic Engineering, Vol.20, No.1, pp.23-30, 1995.
- [5-49] K. Ishii, T. Fujii, T. Ura, "An On-line Adaptation Method in a Neural Network Based Control System for AUV's", IEEE Journal of Oceanic Engineering, Vol.20, No.3, pp.221-227, 1995.

- [5-50]石井和男, 浦環, 藤井輝夫, "ニューラルネットワークによる潜水艇の運動の同定(その2: 学習過程の改良とコントローラ調整への適用)", 日本造船学会論文集, Vol.177, pp.429-435, 1995.
- [5-51]石井和男, 藤井輝夫, 浦環, 能勢義昭, "ニューラルネットワークによる潜水艇の運動の同定(その3: 学習による外乱への適応)", 日本造船学会論文集, Vol.182, pp.469-479, 1997.
- [5-52] K. Ishii and T. Ura, "An adaptive neural-net controller system for an underwater vehicle", Journal of IFAC Control Engineering Practice, Vol.8, pp.177-184, 2000.
- [5-53]K. Tokunaga, T. Furukawa and S. Yasui, "Modular Network SOM: Extension of SOM to the realm of function space", WSOM'03, pp.173-178, 2003.
- [5-54]T. Furukawa, T. Tokunaga, S. Kaneko, K. Kimotsuki and S. Yasui, "Generalized Self- Organizing Maps (mnSOM) for Dealing with Dynamical Systems," International Symposium on Nonlinear Theory and its Applications, pp.231-234, 2004.
- [5-55]T. Furukawa, K. Tokunaga, K. Moroshita and S. Yasui, "Modular Network SOM (mnSOM): From Vector Space to Function Space", International Joint Conference on Neural Networks, 2005.
- [5-56]K. Tokunaga, T. Furukawa, "Nonlinear ASSOM Constituted of Autoassociative Neural Modules", 5th Workshop on Self-Organizing Maps, 2005.
- [5-57] T. Minatohara, T. Furukawa, "Self-Organizing Adaptive Controllers: Application to the Inverted Pendulum", 5th Workshop on Self-Organizing Maps, pp.44-48, 2005.
- [5-58] 西田周平, 石井和男, 古川徹生, 水中ロボットにおける自己組織的行動獲得システム (第一報: 自己組織化マップを用いた運動制御システムの提案), 日本船舶海洋工学会論文集, 第3巻, pp.205-213, 2006
- [5-59] 西田周平, "自律型水中ロボットにおける自己組織的行動獲得システムの開発, 2006年九州工業大学博士論文
- [5-60] 藤井, 浦, "自律海中ロボット「ツインバーガー」の開発", Proc. Techno-Ocean'94, pp.421-426, 1994
- [5-61] Y.Takemura, M.Ishitsuka, S.Shuto, T.Ashikata, K.Ishii, "System Design of an AUV Considering Modularity Concept and Performance Evaluation Through Yawing Control Experiment", Proceedings of IEEE 2nd Inter National Conference on Underwater System Technology: Theory and Applications 2008, CS\_Paper 16\_Shingo.pdf, Bali, Indonesia, 2008,

- [5-62] 石塚誠, “マニピュレータを搭載した自律型水中ロボットの開発”, 九州工業大学博士論文, 2008
- [5-63] S. Nishida, K. Ishii and T. Furukawa, "An adaptive controller system using mnSOM", Proc. of Brain-inspired Information Technology, pp.181-184, 2005.
- [5-64] 西田周平, 石井和男, 古川徹生, "mnSOM を用いたニューラルネットコントロールシステムの開発", 平成17年日本船舶海洋工学会秋季講演会予稿集, pp13-14, 2005.
- [5-65] 西田周平, 石井和男, 古川徹生, "mnSOM を用いた水中ロボットの自己組織的 行動決定システムの開発", SICE システムインテグレーション部門講演会 (SI2005)予稿集, pp655-656, 2005.
- [5-66] 西田周平, 石井和男, 古川徹生, "mnSOM を用いた適応制御システムの水  
中ロボットへの適用", ロボティクスシンポジウム予稿集, pp264-269, 2006.
- [5-67] 西田周平, 石井和男, 古川徹生, "水中ロボットにおける自己組織的行動獲得システム -第 1 報:自己組織化マップを用いた運動制御システムの提案-", 日本船舶海洋工学会論文集 第 3 号, pp.205-213, 2006.
- [5-68] 西田周平, 石井和男, 古川徹生, "モジューラネットワーク型自己組織化マップを用いた水中ロボットのシステム同定", ロボティクス・メカトロニクス講演会2006予稿集, 1P1-E23, 2006.
- [5-69] S. Nishida, K. Ishii, T. Furukawa, "An Adaptive Neural Network Control System using mnSOM", Proc. of Oceans'06 Asia Pacific, FRI-04-4, 2006.
- [5-70] 西田周平, 石井和男, 古川徹生, "モジューラネットワーク型自己組織化マップを用いた水中ロボットの運動制御システム", 第 24 回日本ロボット学会学術講演会予稿集, 3D37, 2006.
- [5-71] S. Nishida, K. Ishii, T. Furukawa, "Adaptability of mnSOM Based Control System to Changing Dynamic Property", Proc. of Brain-Inspired Information Technology, 2006.
- [5-72] S. Nishida, K. Ishii and T. Furukawa, "Development of a Control System for Autonomous Underwater Vehicles using mnSOM", Proc. of SCIS&ISIS 2006, TH-E4 100305, 2006.
- [5-73] S. Nishida, K. Ishii and T. Furukawa, "An Online Adaptation Control System using mnSOM", Proc. of ICONIP 2006, pp.935-942, 2006.
- [5-74] K.Ishii, S.Nishida, Y.Takemura, T.Furukawa, “An adaptive control system using modular network SOM”, APuuRobo2008, 2008

- [5-75] 武村 泰範,首藤 槓吾,西田 周平,石井 和男,古川 徹生:モジューラ型自己組織化マップを用いた水中ロボットのシステム同定(第2報:自律型水中ロボット“Twin-Burger”のダイナミクスへの応用),ロボティクス・メカトロニクス講演会2008 予稿集 長野,2A1-A08(1)-(4.),2008
- [5-76] Y.Takemura, K.Ishii, S.Nishida, T.Furukawa, “Learning Dynamics Property of Underwater Robots Controller System using mnSOM, ~Development of Forward Model Modules ~”, The 9th POSTECH-KYUTECH Joint Workshop 2009
- [5-77]首藤槓吾, “自律型水中ロボットにおける運動制御システムの開発研究”,九州工業大学情報工学部卒業論文, 2007
- [5-78]藤井輝夫, “海中ロボットの知的行動に関する研究”, 東京大学博士論文, 1992

# 謝辞

本研究を遂行するにあたり、熱心なご指導と有益な御助言を賜りますとともに、研究者としての心得を御教授してくださいました九州工業大学大学院生命体工学研究科脳情報専攻 石井和男准教授に心から感謝します。COE マルチタレント英才教育や博士後期課程においての研究活動に対して様々な貴重なご意見を賜りました。九州工業大学大学院生命体工学研究科 脳情報専攻 古川徹生教授、神酒勤准教授、中川秀樹准教授に深く感謝申し上げます。

北九州市立大学国際環境工学部機械システム工学科山本郁夫教授、九州工業大学大学院生命体工学研究科脳情報専攻 古川徹生教授、宮本弘之准教授には本論文の審査にあたり様々な助言を頂きました。本当にありがとうございました。

本研究を進めるにあたり、ロボカップでの研究では、北九州市立大学国際環境工学部情報メディア工学科 Ivan Godler 教授、九州工業大学大学院生命体工学研究科脳情報専攻 石井研究室研究員 Amir Ali Forough Nassiraei 氏には、大変お世話になりました。私が研究活動を6年間も根気よくやれたのは先生方のご指導があったからこそだと思っています。また、水中ロボットの研究や学会活動などにおいて東海大学海洋学部海中システム工学科 渡邊啓介准教授にも非常に公私にわたりお世話になりました。渡邊先生と石井先生から研究の楽しさや厳しさを教えて頂きました。いつか先生方のような研究者になれるように努力したいと思います。

学部4年生から6年間お世話になった石井研究室的メンバーにも感謝したいと思います。西田周平さんをはじめとする諸先輩方の生き活きとした姿は僕たちの憧れの的でした。石井研究室に入っていなかったら、博士まで研究したいという気持ちは出なかったでしょう。本当にお世話になりました。佐藤さんには、学部4年のときからお世話になりました。よく衝突することもありましたが、石井研究室的の仲間として今後もいろいろと相談できる先輩としてこれからもよろしくお願いします。松尾さんは、いつも私のことを励ましてくれる存在でした。研究を投げ出そうとしたときに松尾さんの一言に何度助けられたかわ

かりません。これからも、よろしくお願ひします。園田さんは、年が同じだという事もありいろいろと相談できる友でもありました。これからも同じフィールドで仕事をする事と思いますが相談にのってください。

また、私は同級生が3名もいるという恵まれた環境で博士前期・後期時代を過ごす事が出来ました。同級生の存在は良きライバル関係であり、信頼し、苦楽を智に出来る仲間でもありました。神田篤君は、責任感も強くいつも頼りっぱなしでした。彼に機械の事を相談し、手伝ってもらったこともありました。本当に信頼できる仲間でした。ありがとう。江里口優君とは、水中ロボットの開発や AUV に一緒に参加したりといろいろと共に行動する事がありました。研究にひたむきに向かう姿はお手本にするものがあり、最後まであきらめずに研究に向かう姿は尊敬していました。昨年の同級生で行く高知・和歌山への旅に参加できなかった事が残念です。機会があれば、また同級生4人と旅行でもしましょう。真田篤君とはロボカップの Hibikino-Musashi のメンバーとして5年間一緒に戦ってきました。彼のおかげで5年間もロボカップに携われる事が出来ました。一緒にドイツで飲んだビールの味は一生忘れないでしょう！ありがとう。

後輩のメンバーにもいろいろと支えてもらいました。博士後期の西田祐也君には、後輩の面倒や様々な研究室行事で助けてもらいました。彼の積極的な性格は憧れでした。また、彼のように包容力のある人間性になりたいと思わせる後輩でした。ありがとう。また、北住祐一君には彼が学部時代からの戦友でした。彼とは、ロボカップでの開発の一番つらい時期を一緒に共にしてきたのでいろいろな思い出があります。その中でも、ドイツ大会で一緒にいったブレーメン大会の事が一番印象に残っています。これからもロボカップの研究をやっていくと思うのですが、お互いに2050年まで行きてワールドカップのチャンピオンに勝てるように研究をがんばっていきましょう。ありがとう。

博士前期課程の足利君、久保君、松尾君、首藤君、松井君、原君、山田君、安藤君、小林君、高木君、棚町君、福永君、松岡君、和田君、学部4年生の馬場君、本田君、横田君にも感謝の意を表したいと思います。首藤君や山田君には私生活でもいろいろと遊びにいたりとお世話になりました。ありがとう。

最後に、私の大学生活を経済的にも精神的にも支えてくれた、両親と友人に心から感謝します。