

# 修士論文

Cascaded-ANFIS の FPGA 実装と心臓病データセットを用いた  
性能評価

FPGA implementation of Cascaded-ANFIS and performance  
evaluation using heart disease datasets

---

報告者

学籍番号：1275047

氏名：宇田見 萌衣葵

---

指導教員

星野 孝総 教授

---

令和7年2月18日

基盤工学専攻 電子・光コース

---

# 目次

|       |                                           |           |
|-------|-------------------------------------------|-----------|
| 第1章   | はじめに                                      | 1         |
| 第2章   | <b>ANFIS(Adaptive-Neuro-Fazzy-System)</b> | <b>4</b>  |
| 2.1   | ANFIS とは . . . . .                        | 4         |
| 2.2   | ANFIS の研究紹介 . . . . .                     | 5         |
| 2.3   | ANFIS と他の機械学習との性能比較 . . . . .             | 6         |
| 第3章   | <b>4入力3MF-ANFIS の設計と実装</b>                | <b>9</b>  |
| 3.1   | ANFIS の SoC-FPGA への実装準備 . . . . .         | 9         |
| 3.2   | ANFIS の学習 . . . . .                       | 10        |
| 3.3   | 論理回路の設計 . . . . .                         | 11        |
| 3.4   | 使用するデータセット . . . . .                      | 13        |
| 3.4.1 | アヤメデータセット . . . . .                       | 13        |
| 3.4.2 | 天秤データセット . . . . .                        | 14        |
| 3.5   | 結果と比較 . . . . .                           | 14        |
| 3.5.1 | アヤメデータセットの測定結果 . . . . .                  | 15        |
| 3.5.2 | 天秤データセットの測定結果 . . . . .                   | 16        |
| 3.5.3 | 2つのデータセットによる4入力3MF ANFIS の学習と実装結果 . .     | 19        |
| 3.6   | データセットによる ANFIS の精度 . . . . .             | 20        |
| 第4章   | <b>Cascaded-ANFIS</b>                     | <b>21</b> |
| 4.1   | Cascaded-ANFIS の概要 . . . . .              | 21        |
| 4.2   | Cascaded-ANFIS の性能の評価 . . . . .           | 22        |
| 4.2.1 | 虚血性心疾患(きょけつせいしんしっかん) . . . . .            | 23        |
| 4.2.2 | 心臓病データセット . . . . .                       | 23        |
| 4.2.3 | 学習用データセット . . . . .                       | 24        |
| 4.2.4 | 精度の評価 . . . . .                           | 28        |
| 4.2.5 | 考察 . . . . .                              | 31        |
| 4.3   | 他機械学習との Accuracy の比較 . . . . .            | 32        |
| 4.3.1 | 考察 . . . . .                              | 32        |
| 4.4   | まとめ . . . . .                             | 33        |
| 第5章   | <b>結論</b>                                 | <b>34</b> |
|       | <b>参考文献</b>                               | <b>36</b> |

# 第1章 はじめに

IoTとはモノのインターネットであり、情報、データ、リソースを自動的に生成、収集、計算、共有することで、状況や環境の変化に対応できる、ソフトウェアとハードウェアの融合したネットワークシステムと定義されている [1][2]。

今日ではIoTは効率と生産性を向上するために様々な分野で使用されている。農業分野に関しては農業活動の適切な水位を制御 [3]、パターン分析技術と組み合わせてスマート農業ソリューションを提供 [4]、作物の病気の検出 [5]、など農業分野における学習ベースのツールとIoTの重要性が確立されている。ヘルスケアの分野では医療現場でのスマートフォンのNFC通信とクラウドを活用した電子カルテへの記録 [6] がある。また、文献 [7] ではCPAPとHOTにおける遠隔モニタリング、ベッドサイドでのバイタルデータ取得とシステム連携、AIによる認知症予防システムの研究開発などが紹介されている。上記のようにIoTが様々な分野で活用される中、技術の向上とともに改善すべき課題が発生しており、「スループットの向上」「消費電力の削減」「レイテンシーとジッターの削減」「ネットワークセキュリティの向上」などが挙げられている [8]。

これらの課題を解決する手段としてFPGAを制御デバイスとして採用され始めている。FPGAなどのプログラマブルハードウェアデバイスは、ニューラルネットワークやPID制御のハードウェア実装に適している。クロックに従って回路を実行することから論理回路の再現性が高く、センサから直接データの入出力を行うことで、処理速度が速くなるなどの利点がある。また、順序処理をするCPUと異なり、並列処理が可能なFPGAは処理内容によってはCPUよりも高速な処理が可能であるとElnawawyらは述べている [9]。MuradらはCPUとGPUとFPGAの消費電力を比較検証したが、FPGAは入力処理や複雑なカーネルの処理など、どの処理内容でもCPUと比較して優れており、完全なパイプラインの場合、FPGAはGPUやCPUよりも優れたパフォーマンスを発揮すると述べている [10]。ネットワークのセキュリティでは、かつて安全でないIPリンクを使用して相互に通信していた2つのネットワークが、FPGAベースのBITW-IPSecハードウェアソリューションを用いることで、通常のIPの下にIPSecを階層化し、安全に通信できるようになった [11]。これらの利点から、上記のスループットや消費電力などの課題を解決できると考えられている。

また、近年ではクラウドベンダー各社がコスト削減やリアルタイム性などの特性によってFPGAを採用している。ZDNETの羽野三千世の記事によると、MicrosoftはAzure上のFPGAの性能を披露し、ロシア語で書かれた1440ページの『戦争と平和』をAzureのサーバ上で英語に翻訳した際の処理速度として、CPUのみ有効の際は約20秒かかったのに対してFPGAを有効にした場合は約2.6秒で完了した。また、Microsoft Researchの「Project Catapult」では、サーバとFPGAアクセラレータボードによるクラスタを構成した結果、検索エンジン「Bing」の処理において同様のレイテンシを維持するためのサーバ台数は、おおよそ2分の1となった。このようにFPGAはこれまではCPU単体によって処理され

てきたサーバ管理などを補助することによって、処理の高速化とコスト削減を達成し IoT の分野で大きく貢献している。

ここで、ハードウェア・ソフトウェア協調設計による SoC-FPGA の小型 IoT 医療デバイスの可能性を考えた。ハードウェア・ソフトウェア協調設計とは、ハードウェアとソフトウェアを同時に設計し、それぞれが得意とする処理を担当し効率的かつ最適なシステム全体の性能を実現する設計手法である。上記で記述された FPGA による効率化も CPU と FPGA による協調設計である。CPU 単体ではなく ASIC または FPGA を利用し計算することをヘテロジニアスコンピューティングとも呼称されている。従来小型デバイスとして採用されているマイコンは CPU による順序処理であり、I/O ポートからの入出力には記憶装置を介する必要がある。一方、FPGA は並列処理かつセンサからの情報を直接入出力できるため遅延が短い。また、回路の内容によっては CPU よりも低消費電力で稼働することから小型デバイスの制御デバイスとして優れていると考えられる。

小型 IoT 医療デバイスの必要性としては、日本の高齢化とそれに伴う心疾患の増加が予想される。厚生労働省の人口動態統計（確定数）によれば、2023 年の日本の死亡原因で心疾患は 2 番目に多かった [12]。虚血性心疾患は心疾患の 1 種であり、男性では 60 代以降に多く、女性では 70 代以降に多いとされている。症状としては狭心症と心筋梗塞があり、狭心症は 15 分で症状が治まった結果病院に行かず重症化するケースがあり、心筋梗塞は発症から 1 時間以内に適切な処置を行わなかった結果亡くなる場合がある。どちらも突然死の死因となりうるが、高齢者の場合は初期症状が発生しても年齢によって生じている一時的な疲れや体調不良の症状と似ているため、発生診断が遅れる場合が多い。図 (1.1) は厚生労働省が公開している日本の 65 歳以上の人口推移のグラフである。2025 年には日本の国民の約 30[%] が 65 歳以上とされており [13]、心疾患による死亡者の増加が危惧される。小型 IoT 医療デバイスは増加が予想される高齢者の虚血性心疾患による突然死の件数を予防することができると考えられる。

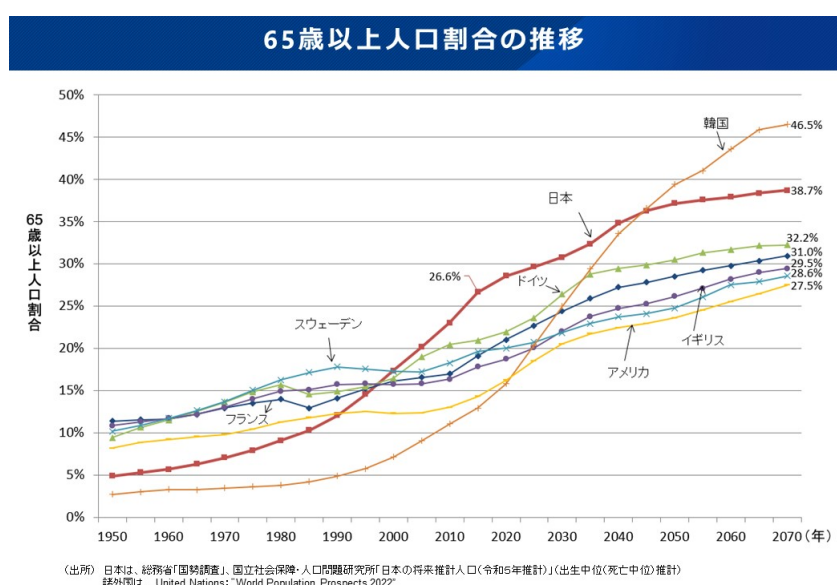


図 1.1: 日本の 65 歳以上の人口推移 (出典：我が国の人口について | 厚生労働省 [13])

小型デバイスで有名な Apple Watch には、心拍数や血中酸素濃度、心電図の測定などの機能が搭載されており、TECH INSIDER の石川 温の記事 (<https://www.businessinsider.jp/article/250124/>)

によれば、当時 50 歳の男性が、Apple Watch に心房細動と忠告されたことをきっかけに病院で診断を受け心臓手術を行った実例がある。小川聡クリニックの記事 (<https://ogawasatoshi-clinic.com/lecture03/lecture1-06.html>) では 2021 年に、当時 57 歳の男性が、Apple Watch で心電図を撮った結果心房細動のサインがでたことをきっかけに来院した。Apple Watch による心電図はとても安定した記録で、クリニックは心房細動と診断した。腕時計型である Apple Watch は日常的に装着されることにより、身体に異常が発生した際の早期発見が可能となる。毎日装着しているデバイスから忠告を受けることにより、自己判断によって病院へ診断を受けに行かなかった結果症状が悪化するという事態を減らすことができると考えられている。このように Apple Watch のような AI 搭載の IoT デバイスが診断する結果により健康をサポートしている。

しかし、Apple Watch など IoT デバイスによる心電図などの測定はあくまでヘルスケアという数ある機能の 1 つであり、それ自体は医療機器ではない。各病状の診断に必要な情報を IoT デバイス単体ですべて習得するには限界がある。そこで、本研究では医療現場と情報を共有し、センサによって習得可能な情報は 24 時間更新し、このようなバイタルデータから即時心臓病を診断する FPGA を制御デバイスとした小型 IoT デバイスの開発と実装を最終の目標とし研究する。

心臓病クラスタリングアルゴリズムとして FPGA に実装する AI に ANFIS(Adaptive-Neuro-Fuzzu-System)を採用する。ANFIS(適応型ニューラルファジ推論システム)は Jang[14] が高木・菅野ファジィ推論[15]を基に作成した人工ニューラルネットワークである。ニューラルネットワークの学習能力とファジィ論理の推論能力を組み合わせることで、複雑な非線形関係をモデル化する能力があり、太陽光発電システムの MPPT 制御[16]や半導体製造技術のエッチング工程における温度管理[17]など、非線形システムのモデリングや制御に使用される研究が多い。ANFIS は特徴として、学習によって処理システムを更新する際、構造は変更せず関数内の定数を変更すれば完了できる。よって、学習結果を更新しながら検証を行うことができる。

本研究の目的は、心臓病データセットを学習した ANFIS を FPGA に実装し、そのクラスタリング性能を検証することで、小型デバイスによる心臓病診断が可能であるかを評価する。

## 第2章 ANFIS(Adaptive-Neuro-Fazzy-System)

### 2.1. ANFIS とは

ANFIS は人工ニューラルネットワーク (ANN) の 1 種である。ANN とは人間の脳を模倣した計算モデルである。ANN はプログラミングからではなく、人間と同様に、適切な学習の経験を通じて学習または訓練される [18]。人間が物体を記憶する際、その物体の特徴（色、形、サイズなど）を抽出し、ニューロンのネットワークを通すことで符号化し、シナプス結合を強化することで記憶をする。目の前の物体が何か判断する際は、視覚野のニューロンが物体の特徴を抽出し統合することでイメージを形成する。イメージを記憶(他ニューロンのネットワーク)と照合して関連する情報を引き出し、判断に使用する。ANN も同様にデータのパターンと関係をニューラルネットワークを通して検出することにより、知識を収集し、判断する。

ANFIS は 1993 年に Jang によって提案された [14]。Guerra の文献内では ANFIS は事前のモデリング知識がなくても、複雑なシステムの入力と出力の間のマッピングを構築することができるアーキテクチャと学習手順であると説明されている [19]。ファジィ推論システムの利点は、不完全な感覚情報を持つシステムを処理する体系的な計算を提供できることだが、時間とともに変化する環境に対処する適応性が不足している。ANFIS はバックプロパゲーションによって学習することにより環境の変化に対応することが可能である [16]。ANFIS はニューラルネットの学習法であるバックプロパゲーションに加えて勾配降下法と最小二乗法を組み合わせた学習を行う。この学習によって高木・菅野ファジィ推論 [15] における If-Then ルールの前件部のメンバーシップ関数と後件部のパラメータを最適化する。ANFIS の学習は RMSE と目標値を比較し、目標値を下回るまで反復を繰り返す。

ANFIS は 5 層のネットワーク構造(図 2.1)で、各層のノードの数は入力変数の数、各入力に関連付けられたメンバーシップ関数 (MF) の数、ルールベースの数、および出力の数に応じて変化する [20]。第 1 層はファジィ化レイヤーと呼ばれ、各入力から MF の生成を行う。第 2 層では if-then ルールに基づいて第 1 層の出力から各ノードの発火強度を出力する。第 3 層では第 2 層の出力を正規化し、第 4 層では非ファジィ化することで、正規化された活性化度を乗算した結果を入力とし線形結合させた結果を出力する。最後に第 5 層では第 4 層の出力を全て総加算することで最終的な結果を取得する [19]。

第 1 層の処理内容である MF はいくつかの種類の関数が存在し、三角関数やベル型関数などがある。本研究では MF にパイ型関数 (pi-shaped membership function) を用いる (式 2.1)。パイ型関数 (以降 pimf) の特徴は、使用する演算の処理が四則演算と比較処理のみで構成されている非線形関数となっている。本論文ではハードウェア記述で設計をする際、浮動小数点は fp16(IEEE 単精度浮動小数点)で浮動小数点を扱う為、指数換算や絶対値で

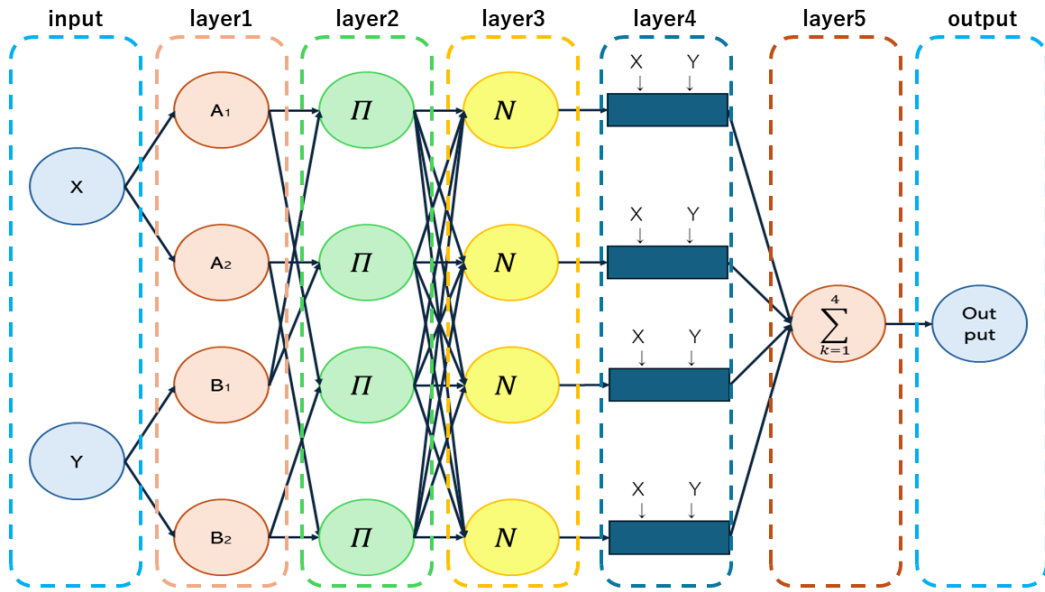


図 2.1: 2 入力 2MF-ANFIS の構造

の計算が複雑になる。よって設計が容易であり、高い精度での演算処理が可能な pimf を FPGA に実装する MF として採用した。この式 (2.1) の場合、学習によって変更されるのは定数  $a \sim d$  である。

$$f(x, a, b, c, d) = \begin{cases} 0 & (x \leq a) \\ 2\left(\frac{x-a}{b-a}\right)^2 & (a \leq x \leq \frac{a+b}{2}) \\ 1 - 2\left(\frac{x-b}{b-a}\right)^2 & (\frac{a+b}{2} \leq x \leq b) \\ 1 & (b \leq x \leq c) \\ 1 - 2\left(\frac{x-c}{d-c}\right)^2 & (c \leq x \leq \frac{c+d}{2}) \\ 2\left(\frac{x-d}{d-c}\right)^2 & (\frac{c+d}{2} \leq x \leq d) \\ 0 & (d \leq x) \end{cases} \quad (2.1)$$

$x$ :input(X,Y) abcd: 各ノードでの定数

第 4 層では式 (2.2) の処理が行われており、非ファジイ化を実行する。

$$\bar{w}_i \times f_i(x, y) = \bar{w}_i \times (k_{i1} \times x + k_{i2} \times y + k_{i3}) \quad (2.2)$$

$i$ : 第 3 層及び第 4 層のノードの番号、 $f_i(X, Y)$ : 入力  $X$  または  $Y$  を代入した  $i$  番目のノードのファジィルール

ANFIS への入力項目の数と第 1 層の MF の数の多さは精度の向上に有効であるという研究結果があるが [21]、各層のノードの数も増加し、設計が困難かつ容量が増加する。

## 2.2. ANFIS の研究紹介

このセクションでは ANFIS が実際にどのような用途で使用されているか、研究の文献を紹介する。ANFIS は前述した通り非線形処理に優れており、太陽光発電システムの MPPT

制御や半導体製造技術のエッチング工程における温度管理など、非線形システムのモデリングや制御に使用される研究が多い。

Yonar らは、大気汚染をモデル化するための遺伝的アルゴリズム (GA)、粒子群最適化 (PSO)、および差分進化 (DE) という3つの一般的なメタヒューリスティック手法を使用して ANFIS の学習をし、従来の ANFIS と性能を比較した [22]。メタヒューリスティック手法とは、ランダムな初期解または母集団を生成することから始まり、次に反復プロセスをループして解または母集団を進化させる。GA、PSO、DE アルゴリズムを使用したトレーニング済み ANFIS と従来の ANFIS によって得られた予測結果は、MSE、RMSE、決定係数の評価項目で、トレーニングされた ANFIS は従来の ANFIS よりも優れた予測値を与えることが観察された。そして一番優れている手法は粒子群最適化 (PSO) であった。

Nosratabadi らは、食料生産を予測するための ANFIS モデルと ANN の MLP モデルを提案した [23]。1961 年から 2017 年までのイランの家畜と農業生産に関連する時系列データが FAOSTAT データベースから収集され、このデータの 70% が ANFIS と MLP のトレーニングに使用され、残りの 30% がモデルのテストに使用された。実験の結果、一般化ベル型メンバーシップ関数が組み込まれた ANFIS モデルが、食料生産の予測において最も低い誤差レベルを持つことが明らかになった。

Huang らは研究において半導体製造技術のエッチング工程における温度管理を、ANFIS によって効率化した [17]。従来の方法では、真空チャンバー内のウェハの温度を安定に保つために、PID コントローラーを利用してヒータープレートの出力を調整していたが、この方法は、ヒーターの PID コントローラーのゲインを調整する経験豊富なオペレーターに依存している。この研究では真空チャンバーの圧力を変更した際の温度を一定に保つために、ヒータープレートを制御するための PWM 信号を ANFIS によって演算し FPGA によって生成する。文献内で提案された方法は、真空チャンバー内の圧力の変動に非常に迅速に応答し、迅速に設定された温度に収束することで、好ましいエッチング速度とエッチングパフォーマンスを確保した。ANFIS を使用した結果は従来の方法よりも速い時間で温度設定の収束が可能で、従来の方法では不可能であった温度の維持を可能としている。

Khaehintung らは ANFIS による太陽光発電システムの最大パワーポイント追跡 (MPPT) のコントローラの設計を提案した [16]。3つの条件で P & O 技術を使用した従来の MPPT コントローラと ANFIS を使用した MPPT コントローラをシミュレーションし、その結果を比較すると、条件1の従来型の沈降時間は0.37秒、条件2では0.69秒、条件3では1.02秒であるのに対して、ANFIS を使用した MPPT コントローラは、条件1の場合、条件2では0.52秒、条件3では0.12秒で定常状態値に達した。実験ではそれぞれのコントローラで発電した場合、ANFIS を使用したコントローラは20秒で最大電力を生成したのに対して、従来のコントローラは約42秒かかった。シミュレーションと実験結果の両方が、提案された ANFIS による MPPT コントローラが従来のコントローラよりも優れていることを示した。

## 2.3. ANFIS と他の機械学習との性能比較

このセクションでは ANFIS と他の機械学習との比較検証を実施している文献の概要と結果をまとめる。

Mokhtarzad らは、干ばつを人工知能技術によって予測するために、ボジュノール気象観

測所の観測データから ANN、ANFIS、SVM(サポートベクターマシーン)で標準化降水指数(SPI)を予測し比較検証を行った。結果として決定係数と K-S 検定の値は SVM、ANFIS、ANN の順で優れており、SVM と ANFIS との性能の差は、決定係数は 0.0048、K-S 検定は 0.0243 であった [24]。

Sarma らはモノのインターネット (IoT) と人工知能 (AI) ベースの監視、制御、意思決定支援の手法を組み合わせて、スマート農業支援システムを設計した。提案されたシステムには、温度記録、空気および土壌水分の継続的なデータ取得を提供するセンサーパックと、AI 意思決定支援システムで使用する植物の葉の近赤外線 (NIR) 画像を取得するカメラが搭載されており、12 種類の植生を特定し、そのうちトマトの葉の 5 つの病気クラスを CNN を使用して分類する。SVM、MLP、TDNN、ANFIS、CNN で Accuracy、交差オーバーユニオン (IOU)、ピクセル精度 (PA) を測定した結果、CNN が最も優れており、次点で ANFIS が優れていた。CNN と ANFIS との性能の差は、Accuracy が 5%、IOU が 3%、PA は 4% であった。また、GLCM、LBP、LBGLCM、GLRLM、SFTA の各機能を使用して、植物の健康状態に関する決定を導き出した結果、同じように CNN が最も優れた各 94% で、次点で ANFIS が GLCM で 87% でそれ以外で 88% であった [25]。Yilmaz らは、隆起パーセントを予測するために ANN の MLP、RBF と MR と ANFIS の性能を比較検証している。結果としては RBF と MLP で構築された ANN モデルはうねり率の予測において ANFIS および MR よりも高いパフォーマンスを示した。ANFIS は MR に対して RMSE と VAF と MAPE と決定係数の評価項目が優れているが、ANN の MLP、RBF より劣っていた [26]。

Armaghani らは、セメントベースのモルタル材料の圧縮強度の予測するシステムの研究において、6 つのモデル入力 (AS、MDA、MK/B、W/B、SP、B/S) と 1 つのモデル出力 (セメントベースのモルタルの CS) で構成される 276 のサンプルデータのセットを学習して圧縮強度予測をする ANN と ANFIS を開発し、性能を比較検証した [27]。結果としてトレーニングおよびテストの予測結果は ANFIS が ANN よりも優れていたが、ANFIS は過学習をしていることが判明し、最適なモデルには ANN が選択された。

Sada らは、AISI 鋼旋削加工における加工応答 (金属除去率と工具摩耗) を予測する際の、ANN と ANFIS の性能を評価した [28]。どちらの技術も、機械加工プロセスの応答を予測する上で優れた能力を示し、ANN および ANFIS の予測の平均誤差は、MRR についてはそれぞれ 6.1% と 11.5%、工具摩耗については 4.3% および 7.2% であることから、ANN がより優秀であるとされた。Sada らによると ANN が効率的であるのは、設計者が ANN モデルをトレーニングするための最適なハイパーパラメータを決定できるためであると考察されている。

上記の研究では ANN や CNN が ANFIS と比較して優れた性能であったが、ANFIS の性能が優れている研究結果も存在する。El-Shafie らは、アスワンハイダム (AHD) におけるナイル川の流入量を月次ベースで予測するために、ANFIS モデルを提案し、ANN モデルと性能を比較している。結果は、ANFIS モデルが、ANN-MLP モデルと比較して、特に極端な流入イベントにおいてより高い流入予測精度を提供できることを示した。ANFIS モデルは、AHD のナセル湖貯水池の水管理において非常に有益であると結論づけた [29]。

CNN との比較でも ANFIS の性能が優れている研究があり、Vatanchi らは、米国のコロラド川の長期的な河川流量予測をするために ANFIS、ANN、BiLSTM、CNN-GRU-LSTM の 4 つの方法で比較した。データの 60%(1921~1981 年) はトレーニングに使用され、40%(1981

～2021年)はモデルのパフォーマンスの評価に使用された。評価項目にはナッシュ・サトクリフ係数がある。ナッシュ・サトクリフ係数は1に近いほど精度が高く、0.7より大きければモデルの再現性が高いとされる。結果として、ANFISがMAE、NRMSE、ナッシュ・サトクリフ係数、決定係数の評価項目で最も優れている。ANFISモデルの出力は、テ일러図に基づく他のモデルよりも実際の観測値に近くなる結果であった[30]。Maharaniらは、金融システムは金融機関の安定性に依存していることから、銀行の倒産確率を予測するためにANFISモデルを使用した。2010年から2021年にかけてインドネシアに上場している42の銀行発行体の財務データを使用してANFISをLSTMとCNNと比較検証した。ANFISは三角関数をMFとし、243のルールをもっている。ANFISは0.1403という最も低いMAPEでLSTMとCNNよりも優れており、銀行の健全性予測において深層学習を上回った[31]。これらの研究文献の性能比較から、ANFISやANNなどの機械学習はクラスタリングするデータセットによって性能が異なることが分かる。本研究におけるアヤメ識別および天秤識別データセットによるANFISの性能は後述する4入力3MFのANFISの研究結果(第3章)によって考察する。

## 第3章 4入力3MF-ANFISの設計と実装

4入力3MFのANFISを学習しFPGAに実装することで、CPUに実装した場合と比較し、処理時間、処理精度、消費電力の観点から、IoTエッジデバイスとしての性能を検証する。

### 3.1. ANFISのSoC-FPGAへの実装準備

FPGAに実装するANFISの論理回路はハードウェア記述言語は「VerilogHDL」、開発は「Quartus II 13.1」によって設計した。VerilogHDLでは値を2進数で扱うため、データセット内の入力項目の値をそのまま扱うことはできず、2進数形式へ変換する必要がある。本論文では論理回路内で演算処理する浮動小数点の実数値を、2進数で16bitの単精度浮動小数点データ型(IEEE754)の形式で扱う(以降fp16)(図(3.1))。32bitのfp32ではなく表現可能な値の範囲が小さくなるfp16を選択した理由は、IoTデバイスの開発において、処理精度よりもコストの軽減を優先した為である。FPGAがCPUにANFISを実装した場合と同様の分類精度を達成した場合、fp16で設計すると、fp32よりもデータのビット幅を半分にできるため、論理回路の容量を大幅に削減できる。また、データセット内の入力項目の値をfp16形式に変更し、FPGAのピン(GPIO)に入出力するために、FPGAはSoC-FPGAを使用することでハードウェア・ソフトウェア協調設計を行う。

FPGAにANFISを設計し実装する際、学習機能を実装するとFPGAの必要容量と利用面積が増加してしまい、小型IoTデバイスとしては課題となる。よって本研究では論理回路の低容化を目的として、学習機能を持たないANFISのモデルのみを実装する。学習はオフライン(MATLAB)で行い、FPGAにプログラムして使用する。

| S    | E    | M     |
|------|------|-------|
| 符号   | 指数部  | 仮数部   |
| 1bit | 5bit | 10bit |

図 3.1: fp16のビットコード

- S(符号) : 0(正)、1(負)
- E(指数) : 2を基数とし、+15のバイアスを加算する(-14から+15)
- M(仮数部) : 仮数が1となったときの小数点以下の数字
- 表現可能な最大値 :  $2^{15} \approx 6.5504 \times 10^4$
- 表現可能な最小値 :  $2^{-14} \approx 6.1035 \times 10^{-5}$

FPGA 内の論理回路へ外部からデータを入力する際、C 言語で記述されたプログラムを CPU で実行し、データセットの値を fp32 形式から fp16 形式に変換する。変換された値は仮想アドレスに格納され、該当する FPGA の GPIO へ出力される。その GPIO から ANFIS の処理モジュールがデータ信号を受け取り、別の GPIO に演算結果を出力する。最後に C 言語プログラムによって、出力側の GPIO から受け取った値を fp32 形式に変換して csv ファイルに格納する。

使用する SoC-FPGA の機種名及び主な機能を表 (3.1) に示す。DE1-SoC は CPU が搭載されており、今回作成した FPGA と CPU 間で行われる値の形式変換プログラムが実行可能であり、かつボード単体で OS(Linux) の機能が使用可能であるため本研究の実験に必要な機能を全て備えているため選定した。

表 3.1: 研究に使用した SoC-FPGA の機能

|                |                    |
|----------------|--------------------|
| 型番             | DE1-SoC            |
| デバイス           | Cyclone V SEA5 SoC |
| LE 数           | 85,000(LE 換算)      |
| ALM            | 32,075             |
| 内部 RAM ビット数    | 3,970Kb            |
| 乗算器            | 174(18 × 19)       |
| プロセッサコア        | Dual ARM@CortexA9  |
| 外部 DRAM(FPGA)  | 64MB SDRAM         |
| 外部 DRAM(HPS)   | 1GB DDR3(HPS)      |
| NTSC ビデオ入力     | 1(ADV7180)         |
| オーディオ入出力       | 24bitCODEC(WM8731) |
| USB2.0Host ポート | 2(HPS)             |
| Earthnet       | 10/100/1G(HPS)     |
| SC-card スロット   | MicroSD            |
| GPIO 拡張バス      | 70I/O              |

## 3.2. ANFIS の学習

本章では、FPGA に ANFIS を実装することを目的とする。初めての実装であるため、ANFIS の学習には機械学習分野で広く使用されているアヤメデータセットを用いることとした。アヤメデータセットは、4つの項目をもつデータセットであるため ANFIS モデルは4入力となった。メンバシップ関数 (MF) の数は最も多く用いられている3とし、4入力 3MF-ANFIS を学習する。

ANFIS の学習には「MATLAB R2017」内のアプリケーションである「Neuro-Fuzzy Designer」を使用した。各 layer で演算する関数をパイ型関数と linear 関数に設定した。パイ型関数を選択した理由は、第3章で記述した通り四則演算と比較演算のみで構成されているからであるが、表 (3.2) で記載されているアヤメデータセットを学習した際の前件部 MF による最小のエラー値が、絶対値や指数関数を使用しない MF の中で最も小さい値であった。

め選択した。学習用データセットは、1つの分類対象に対して4つの入力項目と分類するクラスの正解値を記述した計5項目のデータセットを用意する(表(3.3))。ANFISの出力する値は学習の際に設定するデータセットの各クラスの正解値に基づいて出力され、出力値が最も近い正解値を持つクラスに分類される。また値が正解値に近いほどそのクラスである度合いが高くなるように設定した。

表 3.2: アヤメデータセットを学習した際の前件部 MF による最小のエラー値

| 前件部 MF               | エラー値     |
|----------------------|----------|
| trimf(三角形)           | 1.0951   |
| trapmf(台形)           | 3.7536   |
| gbellmf(一般的なベル型)     | 0.021761 |
| gaussmf(ガウシアン型)      | 0.15818  |
| gauss2mf(ガウシアン2型)    | 0.13541  |
| pimf(パイ型)            | 0.66251  |
| dsigmf(2つのシグモイド型の違い) | 0.026067 |
| psigmf(2つのシグモイド型の積)  | 0.026067 |

表 3.3: 学習する ANFIS のパラメータ

|                          |               |
|--------------------------|---------------|
| 入力項目の数                   | 4             |
| 各入力項目に対する layer1 の MF の数 | 3             |
| MF のタイプ                  | パイ型関数(式 2.1)  |
| 後件部の関数                   | linear(式 2.2) |

MATLAB で学習した結果、ANFIS の構造は図(3.2)のようになった。layer1 のノードの数は12個、layer2、3、4 のノードの数は81個である。layer1 のノードの数は入力項目の数に MF の数を乗算した数。layer2、3、4 のノードの数は MF の数を入力項目の数でべき乗した数が存在する。

### 3.3. 論理回路の設計

図(3.3)は設計された4入力 3MF の ANFIS のデジタル回路の構成図である。x1 から x4 は入力項目であり、分類結果は output として FPGA のピン(GPIO)に出力される。ANFIS の layer1 から layer5 で実行される fp16 の演算処理は Shirazi の文献[32]を参考に設計した四則演算と比較演算の論理回路によって行われる。

各 layer での処理は Kim らの研究[33]を参考にパイプライン方式のデータフロー型計算で処理している。layer ごとのノードに番号を振り当て、クロックによって処理を行うノードを呼び出すように設計されており、layer1 と 4 で使用する定数はメモリからクロックに同期して呼び出す。設計した論理回路の容量や機能などは表(3.4)の通りとなった。ANFIS を搭載していない場合の表(3.5)と比較して、ANFIS を搭載するのに必要な各使用率を算出すると、ロジック使用率は約 11[%]、レジスタ総数は 4212、ブロックメモリー総

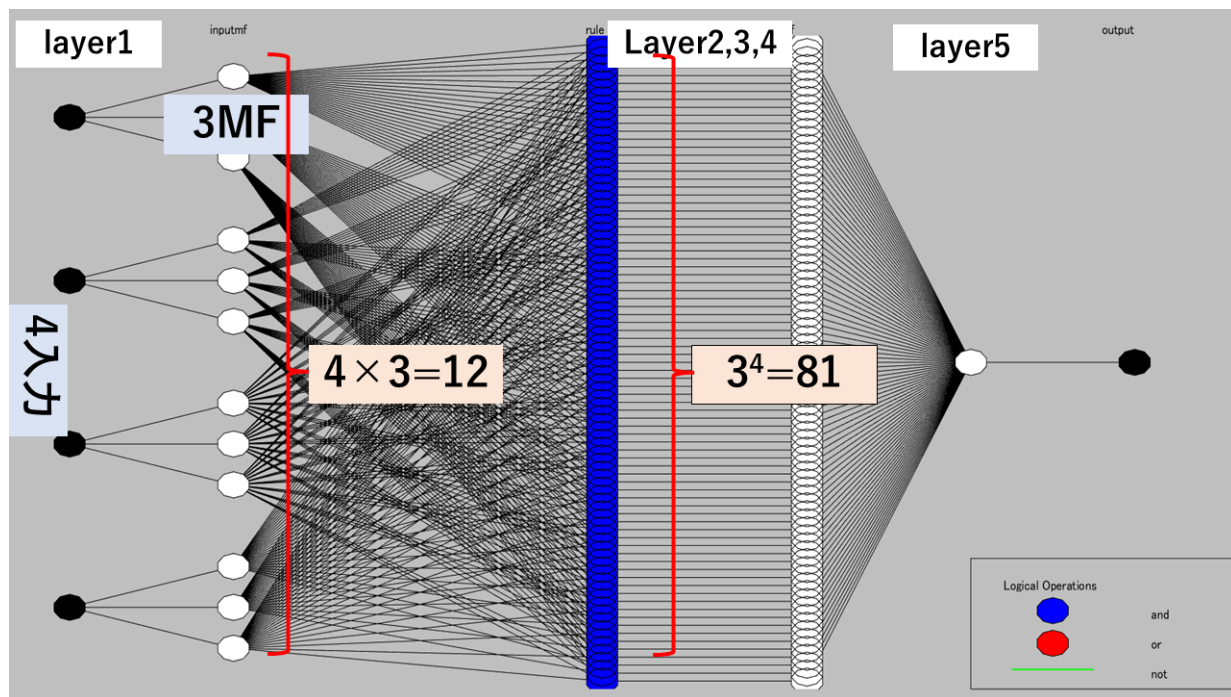


図 3.2: 4 入力 3MF-ANFIS の構造図 (MATLAB 出力)

ビット数は約 1[%]、DSP ブロック合計は 18[%] である。DE1-SoC の使用できるレジスタ総数の容量は公式製品紹介に記載されておらず、レジスタの項目からは判断することはできないが、ANFIS のデジタル回路は容量が小さく、DSP ブロック合計が 5 個の ANFIS で約 90[%] になることから、レジスタ総数を考慮しない場合は 1 台の DE1-SoC に 5 個搭載する余裕があることがわかる。

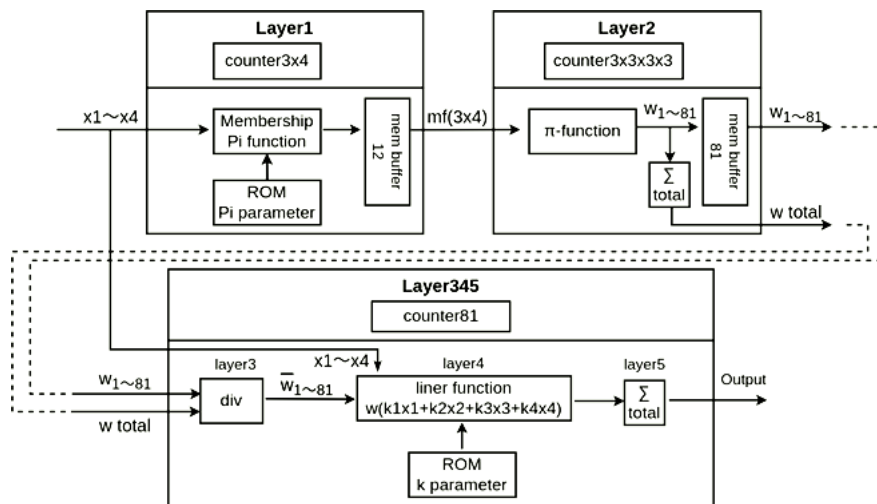


図 3.3: 4 入力 3MF の ANFIS のデジタル回路の構成図

表 3.4: ANFIS デジタル論理回路の機能

|               |                             |
|---------------|-----------------------------|
| クロック周波数       | 50MHz                       |
| ロジック使用率       | 9,038 / 32,070 ( 28 % )     |
| レジスタ総数        | 13245                       |
| ブロックメモリー総ビット数 | 103,262 / 4,065,280 ( 3 % ) |
| DSP ブロック合計    | 16 / 87 ( 18 % )            |

表 3.5: ANFIS を搭載していないデジタル論理回路の機能

|               |                            |
|---------------|----------------------------|
| クロック周波数       | 50MHz                      |
| ロジック使用率       | 5,443 / 32,070 ( 17 % )    |
| レジスタ総数        | 9033                       |
| ブロックメモリー総ビット数 | 95,342 / 4,065,280 ( 2 % ) |
| DSP ブロック合計    | 0 / 87 ( 0 % )             |

### 3.4. 使用するデータセット

4入力 3MF の ANFIS の性能を評価するために、2種類のデータセットを用意し、それぞれで FPGA と CPU で稼働した際の性能を比較検証した。どちらも4つの入力項目と3つの分類するクラスを持つデータセットであり、演算処理をする際の正解値を用いて ANFIS の学習を実行した。

#### 3.4.1. アヤメデータセット

アヤメデータセットの正式名称は「iris-dataset[34]」であり、UCI Machine Learning Repository から入手可能で、機械学習による分類方法の評価に使用される代表的なデータセットである。本研究では4入力1出力で構成されているデジタルであるため使用した。setosa(ヒオウギアヤメ)、versicolor(ブルーカラー)、virginica(ヴァージニカ)の3種の品種のアヤメの、花弁 (petal) とがく片 (sepal) の長さ (cm) と幅 (cm) 計4項目の測定値について、1種につき各50個、合計150個のデータがまとめられている。

- 品種 (3 種) : setosa(ヒオウギアヤメ)、versicolor(ブルーカラー)、virginica(ヴァージニカ)
- サンプル数 : 150 個 (1 種につき 50 個)
- 測定項目 (4 項目) : 花弁 (petal) の幅 (Width) と長さ (Length) とがく片 (sepal) の幅と長さ [cm]
- 正解値 : setosa(100)、versicolor(200)、virginica(300)

### 3.4.2. 天秤データセット

天秤データセットの正式名称は「balance-scale-dataset[35]」であり、UCI Machine Learning Repository から入手可能で、天秤の平衡状態を予測するためのデータセットで、クラスに不均衡があり、分類アルゴリズムの評価や不均衡データ対策の研究に使用されている。単純な構造であることから教育や基礎的な実験に利用されている。本研究ではこのデータセットが4入力1出力で構成されているため使用した。R(右に傾いている)、L(左に傾いている)、B(均衡)の3つの天秤の状態と、左の重み (left-distance)、左の距離 (left-weight)、右の重み (right-distance)、および右の距離 (right-weight) の計4項目のデータが625個記録されている。

- 状態 (3種) : R(右に傾いている)、L(左に傾いている)、B(均衡)
- サンプル数 : 625 個
- 測定項目 (4 項目) : 左の重み (left-distance)、左の距離 (left-weight)、右の重み (right-distance)、および右の距離 (right-weight)
- 正解値 : B(100)、R(200)、L(300)

## 3.5. 結果と比較

アヤメデータセットと天秤データセットを ANFIS で学習し、FPGA と CPU で処理速度、処理精度、消費電力を比較する。FPGA は 50M[Hz] で実装した場合と 100M[Hz] でテストベンチした場合、CPU は周波数とプロセッサのタイプが異なる 2 種類の CPU を用意し実装した場合を測定した。

処理精度の判別方法

- 出力結果が 150 以内であればクラス 1
- 出力結果が 150 より大きく 250 以内であればクラス 2
- 出力結果が 250 より大きければクラス 3

各データセットのクラスを表 3.6 に示す。正解値が 100、200、300 である状態をそれぞれクラス 1、2、3 と設定した。

表 3.6: CPU と FPGA の処理精度の比較 (アヤメ識別)

|       | アヤメデータセット       | 天秤データセット |
|-------|-----------------|----------|
| クラス 1 | setosa(100)     | B(100)   |
| クラス 2 | versicolor(200) | R(200)   |
| クラス 3 | virginica(300)  | L(300)   |

$$(\text{相対誤差}) = \frac{|(\text{正解値}) - (\text{処理結果})|}{(\text{正解値})} \quad (3.1)$$

$$Accuracy = \frac{(\text{正解した分類の回数})}{(\text{分類を実施した回数})} \quad (3.2)$$

Accuracy(式 3.1 の「正解した分類の回数」は、上記の判別方法によって出力結果から判別されるクラスと、処理した対象データのクラスが一致した回数である。

処理速度の判別方法

CPUではANFISのプログラムに、入力項目を読み込んでからlayer5が終了するまでの時間を測定するコードを追加し測定した。FPGAはlayer1が開始してからlayer5が終了するまでの時間をロジックアナライザで測定した(図 3.4)。

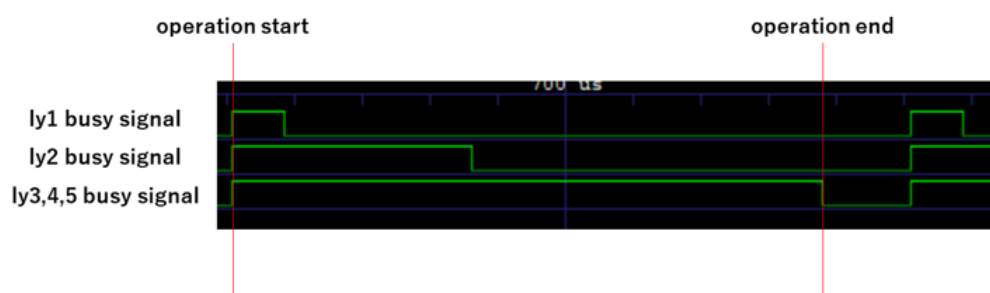


図 3.4: FPGA の測定方法

消費電力の測定方法

CPUは公式サイトで公表されている消費電力を参考資料として判断する。FPGAは本研究で使用するFPGAボードであるDE1-SoCを開発したintel社が公開している、消費電力の見積もりと解析するためのツール「PowerPlay Early Power Estimator (EPE)」を使用し測定する。そして、DE1-SoC全体の消費電力は電源をチェッカーに接続し3分間稼働する間の最大電力と最小電力を測定する。このチェッカーが測定できる最小値は5[W]である。

### 3.5.1. アヤメデータセットの測定結果

処理精度

処理精度を比較するために、CPU(ARM CortexA9)で処理した結果とFPGAで処理した結果を比較した(表(3.7))。正解値と処理結果の差である相対誤差とその相関係数、Accuracyを算出した結果、FPGAはCPUの相対誤差がそれぞれ0.12[%]と0.41[%]で、Accuracyは100[%]であることから、どちらもアヤメを分類するのに十分な処理精度である。よってFPGAによるアヤメ識別はfp16形式での設計が有用であると判断する。

表 3.7: CPU と FPGA の処理精度の比較 (アヤメ識別)

| 制御デバイス       | 相関係数   | 相対誤差の平均 | Accuracy[%] |
|--------------|--------|---------|-------------|
| CPU(float 型) | 1.0000 | 0.12[%] | 100.00[%]   |
| FPGA         | 0.9998 | 0.41[%] | 100.00[%]   |

## 処理時間

FPGA は実装 (50M[Hz]) とテストベンチ (100M[Hz]) の2種類で測定し、layer1 が処理を開始してから layer5 が処理を終了するまでの処理時間を測定した (表 (3.8))。CPU はコア数と動作周波数の異なる2種類を用意し、入力データを読み込んでから layer5 が処理を終了するまでの処理時間を測定するコードを ANFIS のプログラムに追加し測定した (表 (3.9))。ANFIS 論理回路の処理時間では、テストベンチである 100M[Hz] は FPGA に実装した結果の 50M[Hz] の約2分の1の処理時間であることが分かり、実際に FPGA に実装した場合もおおよそ2倍の処理性能での稼働が可能であると考えられる。FPGA に実装した場合の処理速度 (50M[Hz]) を各 CPU と比較すると、2.8G[Hz] の Intel core i7 より遅いが、925M[Hz] の ARM Cortex A9 より速く処理している。しかしながら、それぞれの CPU の動作周波数と FPGA の動作周波数から考えて FPGA の処理速度 87.218[us] は遜色ない性能を持っている。

表 3.8: アヤメ識別をする ANFIS 論理回路の全体の処理時間

| 周波数      | 処理時間       |
|----------|------------|
| 50[MHz]  | 87.218[us] |
| 100[MHz] | 43.602[us] |

表 3.9: CPU へ実装したアヤメ識別をする ANFIS の処理速度

| CPU           | 周波数      | 処理時間        |
|---------------|----------|-------------|
| ARM@CortexA9  | 925[MHz] | 160.431[us] |
| Intel Core i7 | 2.8[GHz] | 12.738[us]  |

## 消費電力量

SoC-FPGA の FPGA 部 (論理回路) と HPS 部 (ARM Cortex A9 プロセッサ) の消費電力、intel 公式製品紹介で記載されている intel core i7 の CPU 消費電力、SoC-FPGA の起動から3分間の最大の消費電力を表 (3.10) に記載する。intel core i7 はコンフィグラブル TDP-up 周波数で動作するときに平均 28[W]、コンフィグラブル TDP-down 周波数で動作するときに平均 12[W] で動作するのに対し、SoC-FPGA は FPGA 部は 0.501[W]、HPS 部を加算して 1.354[W] である。これは TDP-down 周波数で動作する CPU よりも顕著に小さいことが分かる。DE1-SoC(実験ボード)を実際にチェッカーで測定すると、起動時に 7[W] になり次第に 5[W] で静止した。チェッカーが測定できる最小値が 5[W] であることから、SoC-FPGA を搭載した DE1-SoC(実験ボード)は起動後は 5[W] 以下の消費電力で稼働していたことが分かる。このことから FPGA と SoC-FPGA は Intel Core i7 と比較して非常に小さい消費電力で動作することが分かる。

### 3.5.2. 天秤データセットの測定結果

天秤データセットの ANFIS プログラムは、アヤメデータセットの ANFIS プログラムの layer1 と 4 の定数部分を変更し実行した。FPGA は RAM に保存されている定数のデータ

表 3.10: 消費電力の比較 [W](アヤメ識別)

|                       |                                 |
|-----------------------|---------------------------------|
| FPGA 部                | 0.501[W]                        |
| HPS 部 (ARM Cortex A9) | 0.853[W]                        |
| FPGA + HPS 部 (EPE)    | 1.354[W]                        |
| Intel Core i7         | (TDP-down)12[W] , (TDP-up)28[W] |
| DE1-SoC(チェッカー)        | 5~7[W]                          |

を変更している。

### 処理精度

アヤメデータセットのクラス分類に使用した CPU(ARM CortexA9) と FPGA の ANFIS プログラムを、天秤データセットを分類するように変更し実行した結果、表 (3.11) の通りとなった。CPU と FPGA どちらも相関係数は 0.8690 であり、アヤメデータセットよりも相関関係が弱い。また、アヤメデータセットと違い FPGA と CPU 両方の Accuracy が小さく、90[%] を下回っている。

表 3.11: CPU と FPGA の処理精度の比較 (天秤識別)

| 制御デバイス       | 相関係数   | 相対誤差の平均 | Accuracy[%] |
|--------------|--------|---------|-------------|
| CPU(float 型) | 0.8690 | 12[%]   | 88.00       |
| FPGA(fp16)   | 0.8690 | 10[%]   | 87.68       |

天秤データセットを学習した ANFIS の性能を評価するために同じデータセットでクラス分類をしている Shanthamallu の研究文献 [36] の Accuracy を表 (3.12) にまとめた。GrAMME-SG と GrAMME-Fusion のどちらも 90[%] 未満の Accuracy である。このことから本研究で作成した ANFIS の性能は他機械学習と同等の精度であることが分かった。

表 3.12: 各研究による天秤データセットの Accuracy

|                     |        |
|---------------------|--------|
| CPU(ANFIS)          | 88.00% |
| FPGA(ANFIS)         | 87.68% |
| GrAMME-SG([36])     | 79.10% |
| GrAMME-Fusion([36]) | 88.72% |

表 (3.13) は FPGA と CPU 両方の各クラス別の Accuracy である。この表からクラスが右と左が 90[%] を超えているのに対し、均衡のときは約 16[%] と著しく Accuracy が低いことが分かる。均衡の分類結果が全体の精度を低下させていることが判明したが、その原因は均衡が他 2 クラスと異なり、そのクラスとなるための条件が複雑であるからと考えられる。天秤の状態は左右の重りの中心からの距離と重さの乗算した値が大きい方に傾くため、右または左に傾くのはその重りについての入力項目の値が大きい場合が多い。そのため、左右の状態には入力項目の傾向が存在するが、均衡は左右の重りの乗算値が等しい場合で発生する状態のため、入力項目の値が小さいとき、大きいときに関係なく発生する。よって傾向が存在しない均衡は学習が困難となると考えられる。図 (3.5) は MATLAB で

学習した「左の重み」についてのグラフであり、赤色の曲線「in1mf1」が「均衡」について、「in1mf2」が「右に傾く」について、「in1mf3」が「左に傾く」についてのMFである。グラフの横軸は左の重みの値であり、均衡は本来であれば左の重みの大きさによる傾向がなく発生する状態であるが、値の範囲は全体ではなく一部分となっている。このことから、入力項目の値に関係なく発生する傾向を持つクラスはANFISの精度が低くなることが分かる。

次に、精度が高かったアヤメデータセットと精度の低い天秤データセットの、クラス変数に対する各入力項目の相関係数を比較する。表3.14はアヤメデータセット、表3.15は天秤データセットのクラス変数に対する各入力項目の相関係数をまとめたもので、アヤメデータセットはがく片の幅を除いてクラス変数に対して強い相関関係を持っていることに対し、天秤データセットは全ての項目のクラス変数に対する相関関係が弱い。天秤データセットの相関係数が低いのはクラス変数が均衡の状態に対して各入力項目の傾向が存在しないことが原因であると考えられる。このことから、データセットの各項目のクラス変数に対する傾向の有無は相関係数の算出で予測することができ、相関係数の強い項目を多く含むデータセットを学習したANFISは精度が高くなると考える。

FPGAとCPUのAccuracyの差は1[%]であり、アヤメ識別と同様にfp16での設計による精度の低下は許容範囲内であると考えられる。

表 3.13: 天秤データセットのクラス別でのCPUとFPGAの処理精度 [%]

| 制御デバイス       | 均衡    | 右に傾く  | 左に傾く  |
|--------------|-------|-------|-------|
| CPU(float 型) | 16.33 | 97.92 | 90.28 |
| FPGA(fp16)   | 16.33 | 97.92 | 90.28 |

表 3.14: アヤメデータセットの各項目のクラス変数に対する相関係数

|        | クラス変数に対する相関係数 |
|--------|---------------|
| 花卉の幅   | 0.96          |
| 花卉の長さ  | 0.95          |
| がく片の幅  | -0.42         |
| がく片の長さ | 0.78          |

表 3.15: 天秤データセットの各項目のクラス変数に対する相関係数

|      | クラス変数に対する相関係数 |
|------|---------------|
| 左の重み | 0.32          |
| 左の距離 | 0.32          |
| 右の重み | -0.31         |
| 右の距離 | -0.31         |

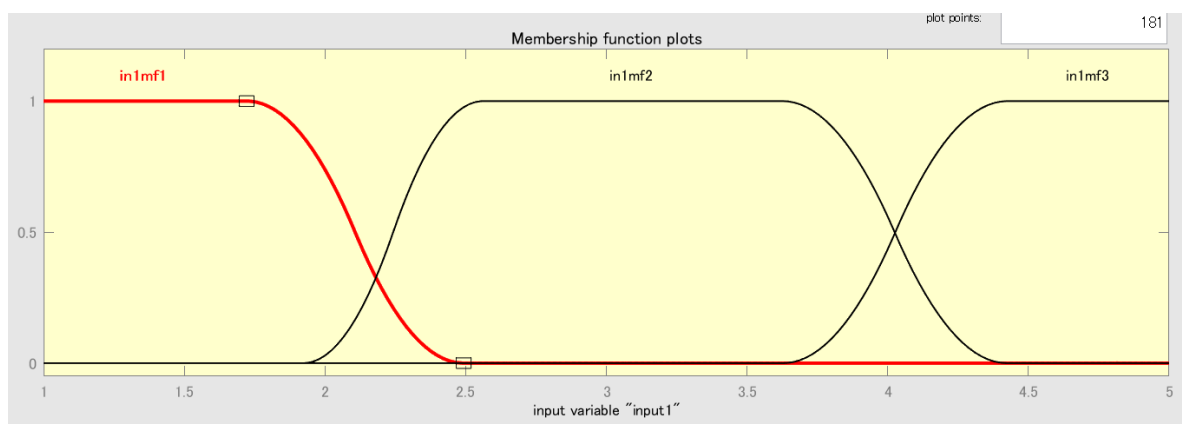


図 3.5: 天秤識別の「左の重み」の MF

### 処理速度

FPGA はアヤメ識別と同様の結果となり、テストベンチ (100M[Hz]) の処理時間は実装 (50M[Hz]) の約 2 分の 1 である (表 (3.16))。そしてアヤメ識別の処理時間に対して 1[us] 以内の処理時間となった。対して CPU の処理時間 (表 (3.17)) は ARM CortexA9 はアヤメ識別の処理時間と大きく異なっており約 90[us] 速くなっており、Intel Corei7 の処理時間も約 4[us] 遅くなっている。ARM CortexA9 は大きく異なっているためその原因を調査中である。クロックに従って稼働する FPGA の論理回路の処理時間は比較対象の 2 種類の CPU よりも安定していることがわかった。

表 3.16: 天秤識別をする ANFIS 論理回路の全体の処理時間 [us]

| 周波数      | 処理時間   |
|----------|--------|
| 50[MHz]  | 87.219 |
| 100[MHz] | 43.613 |

表 3.17: CPU へ実装した天秤識別をする ANFIS の処理速度

| CPU           | 周波数      | 処理時間       |
|---------------|----------|------------|
| ARM@CortexA9  | 925[MHz] | 69.942[us] |
| Intel Core i7 | 2.8[GHz] | 16.174[us] |

### 3.5.3. 2つのデータセットによる 4入力 3MF ANFIS の学習と実装結果

アヤメデータセットと天秤データセットをそれぞれ学習した ANFIS を、VerilogHDL による論理回路と C 言語によるソフトウェアプログラムで開発し、FPGA と CPU に実装して処理精度、処理時間、消費電力を測定し比較した。処理精度はアヤメ識別では FPGA と CPU の Accuracy は 100[%] であることから、どちらもアヤメを分類するのに十分な処理精度であり、天秤識別ではアヤメ識別と異なって Accuracy が低下するが他機械学習と同

等の精度で、FPGA と CPU の Accuracy の差は 1[%] 未満であったことから、fp16 による精度の低下は許容範囲内であった。

処理時間はどちらのデータセットでも FPGA(50M[Hz]) は ARM CortexA9 より早く、intel core17 より遅かった。そして、FPGA はデータセットによる処理時間の差は 1[us] 以内で、2種類の CPU は 1[us] より大きい差が生じている。このことから、処理するデータに関係なく処理時間が安定している FPGA は IoT エッジデバイスとして優れていると考えられる。ただし ARM CortexA9 のデータセットによる処理時間の差はあまりに大きく、原因を調査する必要がある。続いて消費電力では、ARM CortexA9 を HPS 部として含む SoC-FPGA の消費電力量はツール (EPE) によれば FPGA 部と HPS 部を合計して約 1.4[W] で、SoC-FPGA を搭載した DE1-SoC をチェッカーで測定すると起動時の 7[W] を除いて 5[W] 以下であることが確認された。対して intel corei7 の消費電力は公式サイトには平均 12[W](TDP-down) から平均 28[W](TDP-up) と記述されており、消費電力は FPGA と SoC-FPGA が優れていることが分かる。

本研究で作成された ANFIS はデータセットからクラスを分類する用途で設計されたが、接続されたセンサからの入力情報を基に処理を実施する場合、60[Hz] のカメラであれば 16,667[us]、一般的な湿度センサであればであれば 1[s] から 10[s](0.1[Hz]-1[Hz])、温度センサであれば 100[ms] から 1,000[ms](1[Hz]-10[Hz])、小型デバイスである Apple watch Ultra2 は最大 60[Hz] であり、16.667[ms] 以内に情報を処理する必要がある。FPGA に実装した場合の処理時間は 87[us] であり、センサの情報を処理するのに十分な処理速度であることからデータセットの演算処理に限らずセンサからの情報の処理も可能であると考えられる。

処理精度、処理時間、消費電力の 3つの評価項目から、FPGA もとい SoC-FPGA は小型 IoT デバイスとして実用可能であるといえる。

### 3.6. データセットによる ANFIS の精度

第2章で紹介された他機械学習との性能比較された研究文献の紹介では ANFIS の精度が優れている文献と劣っている文献があり、また各データセットによる処理精度に大きく差があることから ANFIS には精度が高くなるデータセットに傾向があり、分類するクラスに対して入力項目の値の傾向がないデータセットでは精度が低くなると考えられる。それぞれのデータセットの各項目のクラス変数に対する相関係数から ANFIS の性能が予測可能であると考察した。

## 第4章 Cascaded-ANFIS

### 4.1. Cascaded-ANFIS の概要

Cascaded-ANFIS(以降 C-ANFIS) は Namal らが提案した新しい最適化アルゴリズムである [37]。このアルゴリズムの特徴は従来の ANFIS の課題である「入力数と MF 数の量によって複雑になる構造と計算」を解決する学習方法と演算方法である。C-ANFIS アルゴリズムは2つの主要モジュール「ペア選択モジュール」「トレーニングモジュール」で構成されている (図 4.1)。

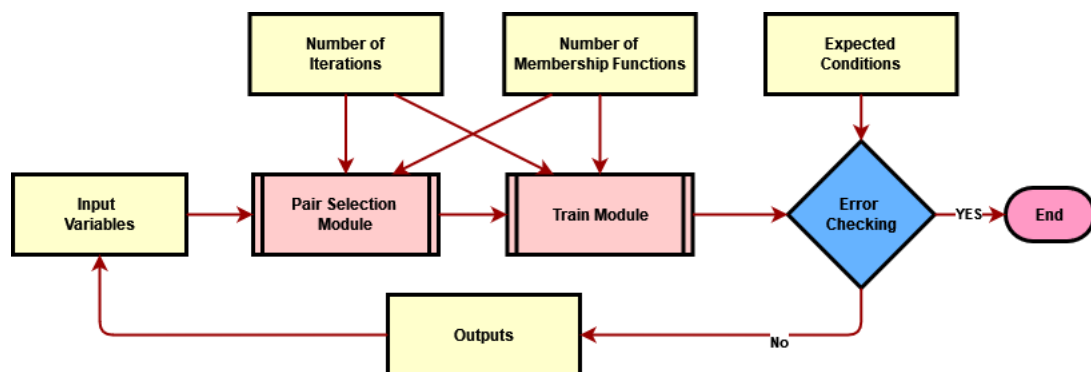


図 4.1: Cascaded-ANFIS のフロー概要図

- ペア選択モジュール

入力項目の数によって生じる課題を解決するモジュールである。このアルゴリズムはすべての機能を使用して堅牢なモデルを構築するため、ノイズの多いデータセットにも有益であるとされている。ペアの選択方法は逐次特徴選択 (SFS) プロセスであり、2入力1出力 ANFIS モデルを使用し、各入力変数の最適な一致を決定する。2入力1出力の ANFIS における RMSE は計算後に保存され、その後、入力項目のすべての組み合わせの中から、最小の RMSE 値を持つ組み合わせ (ペア) が抽出される。

- トレーニングモジュール

ペア選択モジュールで抽出されたペアのトレーニングを行う。ペアの ANFIS の RMSE が生成されると、目標 RMSE 値と比較し、達成できていれば終了し、できていなければ達成するまで反復を繰り返す。

Namal らの性能検証では C-ANFIS と他の2種類の ANFIS である ANFIS-GA と ANFIS-PSO との比較検証を実施し、3種類のデータセットでの処理時間、処理の評価項目 (RMSE、相関係数、テスト時間、トレーニング時間) は C-ANFIS が他2種と比べ優れていた [37]。Cascaded-ANFIS が使用された研究では、Almasri らが性能検証のために長短期記憶 (LSTM)、

SVM、修正適応型ニューロファジー干渉システム (MANFIS)、深層強化学習 (DRL) との比較分析を行った。結果として、C-ANFIS は他の手法と比較して Accuracy が 98.7% という高い検出精度を達成している [38]。Precision、F1-measure などの他評価項目でも C-ANFIS が最も優れていた。

C-ANFIS の構造を図 (4.2) に示す。この図は入力項目が  $N$  個のデータセットを学習した C-ANFIS で、MF の数は 2 個である。各入力項目に対して、2 入力 1 出力の ANFIS が作成され、すべての 2 入力 1 出力 ANFIS ペアの出力結果の平均値が C-ANFIS の最終的な出力結果となる。具体的には、各入力項目は input 層の入力  $X$  に入力され、ペアとして抽出された入力項目が入力  $Y$  に入力される。

C-ANFIS と ANFIS の主な違いは、ネットワークの構造が複雑化しにくい点である。ANFIS の layer2、3、4 におけるノードの数は、MF の数を入力項目数のべき乗で求められるため、2 入力 1 出力の ANFIS で構成されている C-ANFIS は、ノードの数が大きくなりにくいという特徴がある。

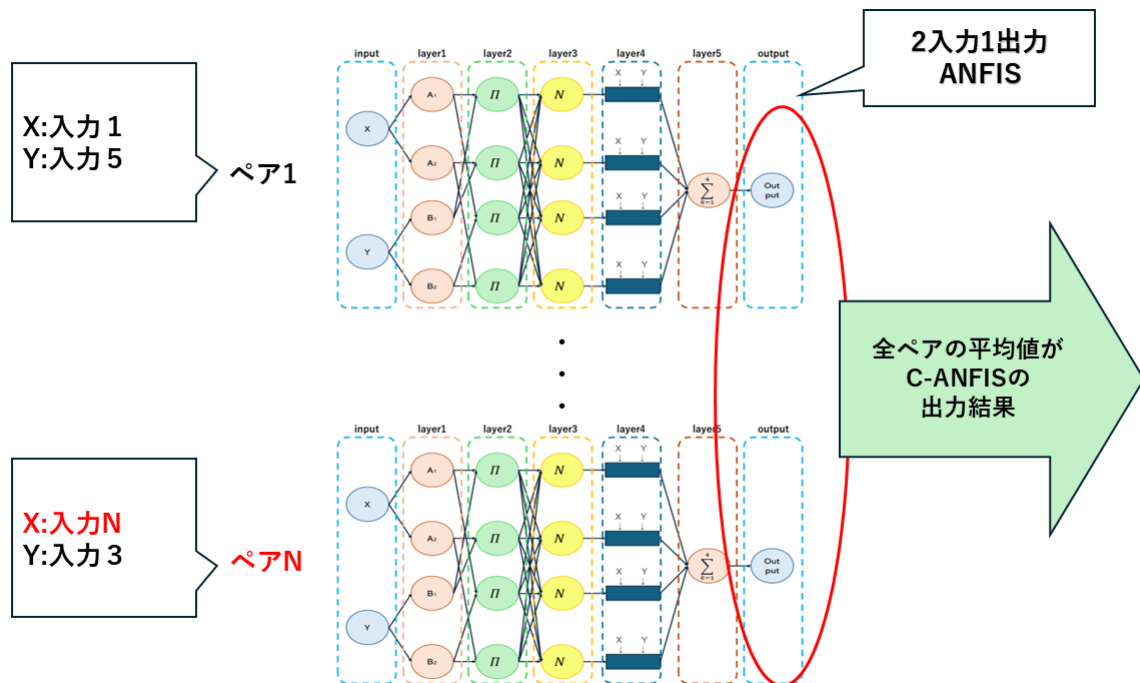


図 4.2: 入力項目が  $N$  個のデータセットを学習した C-ANFIS(2MF) の概要図

## 4.2. Cascaded-ANFIS の性能の評価

4 入力 3MF の ANFIS の FPGA への実装と 2 つのデータセットでの性能の検証と評価を達成したが、ANFIS の学習法による設計には課題がある。ANFIS を使用している各研究文献では、ANFIS の学習に MATLAB を使用しているが、MATLAB での学習には容量の限界が存在し、入力項目が多いデータセット及び MF の数が多い学習が困難である。それに対し、C-ANFIS には入力項目が多いデータセットでも学習が可能である。その理由は、先に述べたペア選択モジュールでの学習において、入力項目が少なく構造がシンプルで、

容量が不足することが無いことに由来する。よって、入力項目が多いデータセットでも学習が可能となる。

また、作成した4入力3MFのANFISの処理構造はクロックに従った順序処理であったが、4入力3MFのANFISよりもニューロンの数が少ないC-ANFISの場合複数のペアを同時に処理することにより、FPGAの並列処理を活用することが期待できる。この章ではC-ANFISの性能を評価するために、心臓病データセット(Heart Disease dataset)をC-ANFISで学習し、その性能を他AIと比較検証する。

#### 4.2.1. 虚血性心疾患(きょけつせいしんしっかん)

虚血性心疾患は、心臓の冠動脈が狭窄や閉塞して心筋に血液が行き渡らず、心筋が酸素不足になることで起こる病気であり、症状には狭心症と心筋梗塞が存在する。診断方法には「心電図検査」、「エコー検査」、「レントゲン検査」、「心臓カテーテル検査」があり、治療方法には「薬物治療」、「カテーテル治療」、「バイパス治療」がある。

虚血性心疾患として代表的な心筋梗塞は、冠動脈の途絶により心筋壊死が生じる病態であり、血行再建術の進歩により、急性期死亡率が著しく減少したが、慢性期に左室リモデリングを生じた慢性心不全症例の増加から生命予後は改善されても、健康寿命は改善されていないと指摘されている[39]。

健康寿命とは「心身ともに自立して健康的に生活できる期間」を指し、世界保健機関(WHO)が2000年に提唱した健康指標である。今後の日本では少子高齢化によって介護を担う人材の不足が考えられることから、自立して生活することが求められる。よって心筋梗塞が発生しないように「虚血性心疾患(心筋梗塞)の予測と警告」が必要であり、小型デバイスによる24時間の簡易診断は健康寿命の延長に貢献できると考える。

#### 4.2.2. 心臓病データセット

心臓病データセット(Heart Disease dataset)[40]はUC Irvine Machine Learning Repositoryで公開されているデータセットであり、心臓病の1種である虚血性心疾患について、医療機関で血管造影検査を受けた患者の14個の項目が記録されている[41]。血管造影検査とは、カテーテルと呼ばれる細い管を目的の血管に挿入して造影剤を注入し、X線撮影で血管の状態を調べる検査である。

この14個の項目を、(a)(b)(c)の3種類のラベルに分類し、表(4.1)に示す。

- (a)：状態が変化しない、変化が予測可能な項目
- (b)：変化の可能性がある、定期的な検査と更新が必要な項目
- (c)：デバイスでの観測が可能である項目

血管狭窄(けっかんきょうさく)は、血管の内径が狭くなる状態を指し、狭窄が進行すると血流が制限され、心臓や脳などの重要な臓器に影響を及ぼし、心筋梗塞や脳卒中などのリスクが高まる。このデータセットでは血管狭窄が直径の50[%]以上の場合(numが1以上の場合)を心臓病(虚血性心疾患)であると判断する。

表 4.1: 心臓病データセットの入力項目と分類されたラベル

| 項目名      | 項目の概要                                                 | ラベル |
|----------|-------------------------------------------------------|-----|
| age      | 年齢                                                    | a   |
| sex      | 性別 (1: 男性, 0: 女性)                                     | a   |
| cp       | 胸痛の種類                                                 | b   |
| trestbps | 安静時の血圧 (入院時)                                          | b   |
| chol     | 血清コレステロール                                             | b   |
| fbs      | 空腹時の血糖値が 120mg/dl より (1: 大きい, 0: 小さい)                 | b   |
| restecg  | 安静時の心電図の結果 (0: 正常, 1: ST-T 波の異常, 2: 左室肥大の可能性又は明確な証拠有) | c   |
| thalach  | 最大心拍数                                                 | c   |
| exang    | 運動時に狭心症を (1: 誘発した, 0: 誘発していない)                        | b   |
| oldpeak  | 運動によって誘発された ST 低下 (安静時と比較)                            | c   |
| slope    | ピーク運動時の ST 部分の傾斜が (1: 上向き, 2: 平坦, 3: 下向き)             | c   |
| ca       | 心臓の主要な血管が透視検査によって色付けされた数                              | b   |
| thal     | サラセミアが (3: 正常, 6: 固定欠損, 7: 可逆性欠損)                     | b   |
| num      | 血管狭窄が直径の 50[%](0: 未満, 1 以上: 以上)                       | b   |

- 状態 (5 種) : 0(心臓病では無い)、1-4(心臓病の段階)
- サンプル数 : 303 個
- 測定項目 (14 項目): 上記

#### 4.2.3. 学習用データセット

心臓病データセットの状態は項目「num」の 0(心臓病では無い)と 1~4(心臓病の段階)である。よって心臓病の段階を表す項目 num から心臓病の有無を表す項目「existence」を作成し、クラス変数が num の場合と existence の場合で Accuracy に変化があるかを確認する。

また、C-ANFIS は多項目データセットの学習が可能であるが、入力項目と同じ数のペアが作成され、2 入力 1 出力の ANFIS の数が増える。入力項目の数が多いと学習の時間がかかる。そこで、入力項目の数を削減した場合の診断性能を比較検証し評価する。心臓病データセットの各項目の相関係数を算出し (図 4.3)、下記の 5 つの方法で項目数を 6 個に削減したデータセットを作成する。これらのデータセットを用いることで、項目数を削減をしていない基のデータセット (心臓病データセット) と比較する。

1. データセット 1: "num" と "existence" を分類するクラスとして、これらに対して正の相関が強い項目を 6 つ選択しデータセットを作成した (表 4.2)。
2. データセット 2: "num" と "existence" を分類するクラスとして、これらに対して相関の絶対値が強い項目を 6 つ選択しデータセットを作成した (表 4.3)。
3. データセット 3: ヒートマップから他の項目に対して 1 様ではない相関を持ち、互いに強い相関を持つ項目の場合は片方のみを選択しデータセットを作成した (表 4.4)。
4. データセット 4: 心臓病データセットを主要主成分分析し、PC1 と PC2 に対する寄与率が大きい順で 6 つの項目を選択しデータセットを作成した (表 4.5)。

## 5. データセット 5: ラベルが (a) と (c) の項目のみで構成した。

表 (4.2)～表 (4.5) はデータセット 1 からデータセット 4 に選択された 6 個の入力項目と、判断基準に使用した相関係数の値や寄与率である。第 3 章の考察より、入力項目のクラス変数に対する相関係数の大きさは ANFIS の性能に影響があると考え、データセット 1 と 2 はクラス変数に対する相関係数が強い項目を選び、2 つのデータセットに含まれる項目で異なる項目は 1 個のみであった。データセット 3 は項目間の相関係数が精度に関係あるかを確認するために、互いに相関係数が強くない項目を選択した。データセット 5 は心臓病の診断および警告を行うためのエッジデバイスを想定して、デバイスで取得可能な情報 (a)(c) のみで診断する場合の精度を評価するために作成した。

学習は「MATLAB R2022b」で行い、過学習を避けるためにトレーニングデータはデータセットの 80[%]、テストデータは 20[%] をランダムに選択し分割したものを使用する。さらに、C-ANFIS と ANFIS の性能を比較するために、各データセットでの ANFIS の学習を行う。しかし、心臓病データセットは学習できる容量を超える為、データセット 1 から 5 のみ ANFIS で学習を行う。また、ANFIS は学習容量が不足するため、C-ANFIS とは異なり MF の数は 2 で学習する。C-ANFIS と ANFIS で epoch 数が異なるのは、C-ANFIS の場合 epoch 数が 100 回未満でエラー値が減少しなくなるが、ANFIS では 100 回以上でも減少するため、機械学習法の確実な性能を比較するために回数を増加した。

- トレーニングデータ: データセットの 80[%]
- テストデータ: データセットの 20[%]
- MF の数: 3(C-ANFIS), 2(ANFIS)
- MF のタイプ: パイ型関数 (式 2.1)
- epoch: 100(C-ANFIS), 300(ANFIS)

表 4.2: クラスとの正の相関が大きいデータ項目の相関値 (データセット 1)

|           | cp(b) | exang(b) | oldpeak(c) | slope(c) | ca(b) | thal(b) |
|-----------|-------|----------|------------|----------|-------|---------|
| num       | 0.41  | 0.4      | 0.5        | 0.38     | 0.52  | 0.51    |
| existence | 0.41  | 0.43     | 0.42       | 0.34     | 0.46  | 0.53    |

表 4.3: クラスとの相関の絶対値が大きいデータ項目の相関値 (データセット 2)

|           | cp(b) | exang(b) | oldpeak(c) | thalach(c) | ca(b) | thal(b) |
|-----------|-------|----------|------------|------------|-------|---------|
| num       | 0.41  | 0.4      | 0.5        | -0.42      | 0.52  | 0.51    |
| existence | 0.41  | 0.43     | 0.42       | -0.42      | 0.46  | 0.53    |

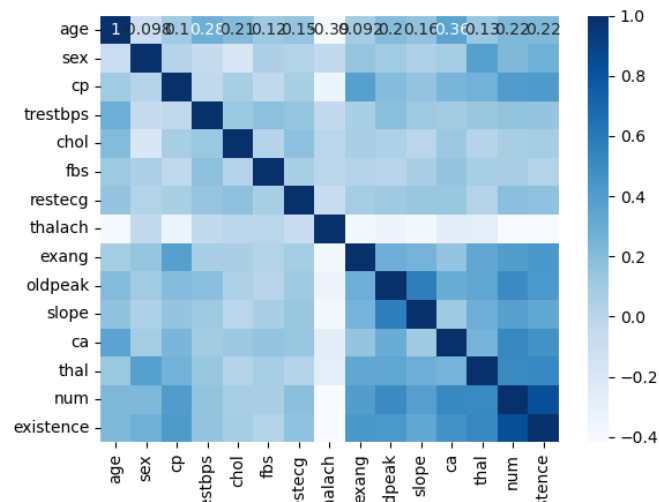


図 4.3: 心臓病データセットの各項目の相関を表すヒートマップ

表 4.4: 他項目との相関値 (データセット 3)

|         | age(a) | thalach(c) | thal(b) | ca(b)   | slope(c) | exang(b) |
|---------|--------|------------|---------|---------|----------|----------|
| age     |        | -0.3938    | 0.1263  | 0.3611  | 0.1618   | 0.0917   |
| thalach |        |            | -0.2719 | -0.2669 | -0.3856  | -0.3781  |
| thal    |        |            |         | 0.2547  | 0.2909   | 0.3347   |
| ca      |        |            |         |         | 0.1132   | 0.1510   |
| slope   |        |            |         |         |          | 0.2577   |
| exang   |        |            |         |         |          |          |

表 4.5: PC1 と PC2 の寄与率 (データセット 4)

|     | age(a) | sex(a)  | chol(b) | trestbps(b) | oldpeak(c) | thal(b) |
|-----|--------|---------|---------|-------------|------------|---------|
| PC1 | 0.1995 | 0.1253  | 0.0619  | 0.1188      | 0.3181     | 0.3092  |
| PC2 | 0.4604 | -0.4466 | 0.4297  | 0.3908      | 0.0176     | -0.2370 |

#### 4.2.4. 精度の評価

各データセットを学習した C-ANFIS の処理性能を評価するために、下記の評価項目を算出する。A は演算結果、P は正解値である。C-ANFIS および ANFIS のクラス分類結果は、演算結果に最も近い正解値を持つクラスに分類されたとする。クラス変数が num と existence のどちらの場合でも、正解値が 0 であれば心臓病ではない (=Negative)、正解値が 1 以上であれば心臓病である (=Positive) と判断する。

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (4.1)$$

$$\text{決定係数 } (R^2) = \frac{S_R}{S_T} \quad (4.2)$$

$$S_R = \sum_{i=1}^n (X_i - \bar{A})^2 \quad (4.3)$$

$$S_T = \sum_{i=1}^n (A_i - \bar{A})^2 \quad (4.4)$$

$$RMSE = \sqrt{(A - P)^2} \quad (4.5)$$

$$Recall = \frac{TP}{TP + FN} \quad (4.6)$$

$$Precision = \frac{TP}{TP + FP} \quad (4.7)$$

TP: True Positive, TN: True Negative, FP: False Positive FN: False Negative

各データセットのテストを実施した後、上記の評価項目を算出した結果が表 (4.6) から表 (4.11) である。各データセットの C-ANFIS 及び ANFIS は、クラス変数が num の精度が高い場合もあれば existence のほうが高い場合もある。このことからクラス変数による精度の向上はないと考えられるが、基データである心臓病データセットの C-ANFIS の精度は num よりも existence の方が高い。

ANFIS は MF が 2 個であるためか、RMSE が同じデータセットで学習した C-ANFIS と比較して大きく、クラス変数が num の場合のデータセット 2、3、4、5 の ANFIS は RMSE が 40 を超えおり、クラス変数が existence の場合でも ANFIS の RMSE はデータセット 1 を除いて 10 を超えている。しかし、RMSE の値が大きい場合でも Accuracy は低くならず、データセット 2 では num(ANFIS) は existence(ANFIS) よりも RMSE が大きい Accuracy が高い。このことから、RMSE が大きくなる要因として、正解値に対して大きく誤差が生じるデータが存在するが、分類結果が FP または FN の時に起こる現象であるため TP もとい TN の時の出力値は正解値と近く、正解率である Accuracy には影響が少ないと考える。

C-ANFIS と ANFIS の性能の比較では、データセット 1、2、5 では C-ANFIS の性能が高いが、データセット 3 と 4 は片方もしくは両方のクラス変数で ANFIS の精度が高い。データセットによって ANFIS と C-ANFIS の性能が一意でないことがわかる。そして、クラス

変数に対する相関係数の大きい順で選択した項目で構成されているデータセット1と2のANFISの精度は高くないことから、第3章での考察とは異なる結果となっている。

続いて、C-ANFISの学習で、項目数を削減することにより基のデータセット(心臓病データセット)と比較して性能が減少または増加するかを比較し評価する。num(C-ANFIS)のAccuracyが最も高いのはデータセット3で値は0.7705、existence(C-ANFIS)のAccuracyが最も高いのはデータセット2と心臓病データセットで値は0.8852である。図(4.4)にクラス変数がnumの場合のデータセット3と心臓病データセットのC-ANFISの各評価項目を示す。データセット3の性能は心臓病データセットと比較して、Accuracy、RMSE、Precision、 $R^2$ が優れており、recallのみ劣っている。項目数を削減したデータセット3の全体的な評価項目が項目数を削減していない心臓病データセットよりも優れていることから、C-ANFISの学習及び演算において、本来はコスト削減のために実行される項目数の削減が、より性能の高いC-ANFISの学習手段となる可能性がある。

図(4.5)にクラス変数がexistenceの場合のデータセット2と心臓病データセットのC-ANFISの各評価項目を示す。データセット2と心臓病データセットはクラス変数がexistenceのとき、C-ANFISのAccuracyの値が等しく、その値もデータセット3のnum(C-ANFIS)のAccuracyより0.1147大きく、心臓病の有無の診断精度がより高いことがわかる。データセット2の性能は $R^2$ とrecallが優れており、心臓病データセットはRMSEとPrecisionが優れている。どちらも2つの評価項目が優れており、診断精度(Accuracy)が等しいことから、特徴量選択をすることは、必ずしも精度の低下を招くわけではないことが分かった。

しかし、心臓病の警告をするエッジデバイスとして、心臓病を見逃してはならない。Precisionは心臓病であると診断した中で実際に心臓病であった割合、recallは心臓病であるデータから診断で発見することができた心臓病患者の割合である。したがって、心臓病の診断においてAccuracyが等しい場合、比較しなくてはならないのはPrecisionとrecallとなる。この場合求められるのは誤診の少なさよりも心臓病を発見できた割合と考えると、心臓病の警告をするエッジデバイスに搭載するC-ANFISとして優れているのはデータセット2で学習したC-ANFISとなる。

入力項目をエッジデバイスのみで測定可能、または測定不要な(a)(c)のラベルの項目のみで構成した、データセット5の性能はデータセット1、2と比較してAccuracyが低く、recallの値も小さい。このことから、心臓病(虚血性心疾患)の診断には心電図で得られる情報(c)の他に、血圧や血糖値、血管などの情報(b)も必要であることが分かった。したがってデバイスによる診断には病院などの専門機関との連携が必須である。

C-ANFISとして総合的に優れているのはクラス変数がexistenceの心臓病データセットを学習したC-ANFISであるが、前述した心臓病の警告をするエッジデバイスのAIとして優れているのは、クラス変数がnumの心臓病データセットを学習させたC-ANFISである。同じクラス変数では全体的な評価項目はデータセット3のC-ANFISのほうが高く、クラス変数がexistenceの心臓病データセットのほうがrecall以外の項目すべてが優れているが、サンプルデータの中から心臓病を発見した割合(recall)が最も高いのはクラス変数がnumの心臓病データセットであった。

表 4.6: データセット 1 の性能

|           | num(C-ANFIS) | existence (C-ANFIS) | num(ANFIS) | existence(ANFIS) |
|-----------|--------------|---------------------|------------|------------------|
| Accuracy  | 0.7377       | 0.7049              | 0.5738     | 0.6721           |
| RMSE      | 1.003        | 0.5268              | 7.0372     | 3.6126           |
| Precision | 0.8333       | 0.9375              | 0.6818     | 0.7500           |
| recall    | 0.625        | 0.4688              | 0.4412     | 0.5625           |
| $R^2$     | 0.3607       | 0.2431              | 0.2582     | 0.1321           |

表 4.7: データセット 2 の性能

|           | num(C-ANFIS) | existence (C-ANFIS) | num(ANFIS) | existence(ANFIS) |
|-----------|--------------|---------------------|------------|------------------|
| Accuracy  | 0.7541       | 0.8852              | 0.7377     | 0.6885           |
| RMSE      | 1.2841       | 0.3590              | 46.0432    | 19.4691          |
| Precision | 0.6944       | 0.9630              | 0.8966     | 0.8095           |
| recall    | 0.8621       | 0.8125              | 0.6667     | 0.5313           |
| $R^2$     | 0.1673       | 0.6119              | 0.0764     | 0.1709           |

表 4.8: データセット 3 の性能

|           | num(C-ANFIS) | existence (C-ANFIS) | num(ANFIS) | existence(ANFIS) |
|-----------|--------------|---------------------|------------|------------------|
| Accuracy  | 0.7705       | 0.6066              | 0.7377     | 0.6557           |
| RMSE      | 1.1454       | 0.6273              | 47.1531    | 12.0569          |
| Precision | 0.7273       | 0.7778              | 0.9063     | 0.6333           |
| recall    | 0.8276       | 0.2414              | 0.6905     | 0.6552           |
| $R^2$     | 0.2489       | 0.0634              | 0.1065     | 0.0968           |

表 4.9: データセット 4 の性能

|           | num(C-ANFIS) | existence (C-ANFIS) | num(ANFIS) | existence(ANFIS) |
|-----------|--------------|---------------------|------------|------------------|
| Accuracy  | 0.6885       | 0.6066              | 0.5574     | 0.7049           |
| RMSE      | 1.2769       | 0.6273              | 46.4639    | 12.6344          |
| Precision | 0.6471       | 0.6923              | 0.8696     | 0.7619           |
| recall    | 0.7586       | 0.3103              | 0.4545     | 0.5517           |
| $R^2$     | 0.1544       | 0.0511              | 0.0034     | 0.1728           |

表 4.10: データセット 5 の性能

|           | num(C-ANFIS) | existence (C-ANFIS) | num(ANFIS) | existence(ANFIS) |
|-----------|--------------|---------------------|------------|------------------|
| Accuracy  | 0.6452       | 0.6230              | 0.6452     | 0.5806           |
| RMSE      | 1.5448       | 0.6222              | 59.0605    | 16.3006          |
| Precision | 0.6061       | 0.6923              | 0.9259     | 0.5714           |
| recall    | 0.6897       | 0.3103              | 0.5556     | 0.4138           |
| $R^2$     | 0.0132       | 0.0537              | 0.0366     | 0.0221           |

表 4.11: 心臓病データセットの性能

|           | num(C-ANFIS) | existence (C-ANFIS) |
|-----------|--------------|---------------------|
| Accuracy  | 0.7377       | 0.8852              |
| RMSE      | 1.2720       | 0.3388              |
| Precision | 0.6905       | 1.0000              |
| recall    | 0.9063       | 0.7308              |
| $R^2$     | 0.1319       | 0.6090              |

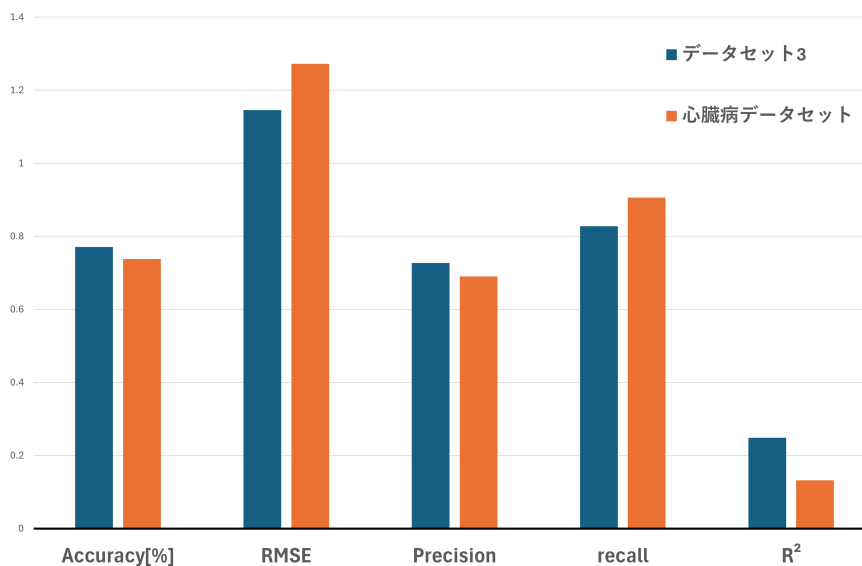


図 4.4: 項目数の削減による基データセットとの性能比較 (num)

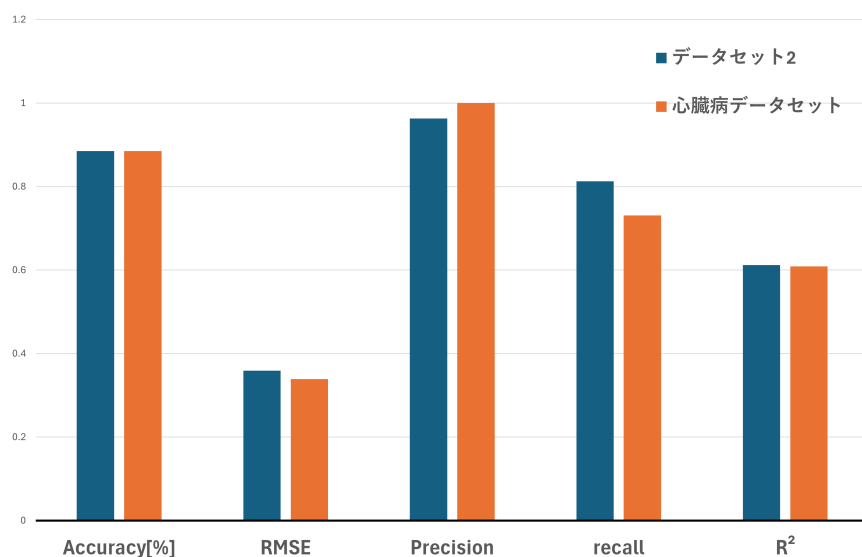


図 4.5: 項目数の削減による基データセットとの性能比較 (existence)

#### 4.2.5. 考察

本章では C-ANFIS の性能の評価をして、容量の削減に向けたペア数 (入力項目数) 削減、クラス変数による精度の検証、同じデータセットで学習した場合の ANFIS との性能比較

を行った。その結果、各データセットによって性能が高いクラス変数は異なり、C-ANFISとANFISどちらが性能が明確に優れていると検証できるデータセットはなかった。また、Accuracyは低いが高recallが高いなど、課題が残る。

表(4.6)から表(4.11)の結果から考察をすると、項目数を削減していない心臓病データセットの性能は、項目数を削減したデータセット1から5のC-ANFISおよびANFISの結果と比較して、各評価項目の性能が高く、項目数を削減してANFISで学習するよりも性能が高いことが示された。項目数を削減したC-ANFISの性能において、データセット2(existence)とデータセット3(num)は心臓病データセットの性能と同等又は優れていることから、性能を低下させずに入力項目の数を削減することは可能ではあるが、項目数を削減するための条件が不明慮であるという結果となった。クラス変数に対して相関係数の正の値または絶対値が大きい順に項目を選択したデータセット1と2は、選択された入力項目の違いは1個のみであるが、データセット1がnumの精度が高いことに対してデータセット2はexistenceの精度の方が高い。

データセット3はクラス変数及びその他の項目に対して相関係数が一様でない項目を選択することによって、相関係数の精度に対する関係性を検証するために作成したが、numでは心臓病データセットよりも高いがexistenceの精度は低い。これらの結果から、クラス変数との相関係数はC-ANFISの精度に影響が小さいと示された。

第3章ではクラス変数に対する相関係数が大きい入力項目を多く含むデータセットは、ANFISの性能が高くなる傾向があると考察した。データセット1と2はクラス変数に対する相関係数が大きい項目を選択したが、表4.2と表4.3からわかる通り各入力項目の相関係数の絶対値は0.41から0.53である。アイリスデータセットは平均して0.7775であるのに対してデータセット1は0.4533と0.4317。データセット2は0.46と0.445である。MFの数が2個であることも第3章との性能の違いがあると考えられるが、データセット3と4のANFISはクラス変数によってはデータセット1と2の性能よりも高い。したがって、他の条件でデータセットを作成し、検証する必要がある。

### 4.3. 他機械学習との Accuracy の比較

本論文で作成したC-ANFISの精度を、他の機械学習で分類した精度[42][43][44][45][46][47]と比較し、C-ANFISの有用性を検証した。表4.12は各機械学習のAccuracyと、心臓病データセットを学習したC-ANFISのAccuracyである。他機械学習のAccuracyは80[%]を超えていることから、クラス変数が基のnumのC-ANFISは精度が低い。しかし、クラス変数がexistenceの場合、C-ANFISはAccuracyにおいてNeural networks ensemble [45]、PSO + ELM [47]、SRLPSO + ELM [47]に次いで4番目の性能であった。

#### 4.3.1. 考察

PSO-ELMとSRLPSO + ELMはクラス変数がexistenceの心臓病データセットを学習したC-ANFISよりも性能が高かった。PSO-ELMは粒子群最適化アルゴリズム(PSO)と極限学習機(ELM)を組み合わせたモデルであり、SRLPSOは自己調整学習をするPSOである。C-ANFISとの性能の違いは学習方法によると考えられる。PSOとは、Particle Swarm Optimizationの略称である。鳥の群れや魚の群れの動きから着想を得た最適化アルゴリズム

表 4.12: 心臓病データセットを学習した各機械学習の Accuracy

| Method                                      | Accuracy(%) |
|---------------------------------------------|-------------|
| Naive Bayes [42]                            | 81.48       |
| C4.5 [42]                                   | 81.11       |
| BNND[42]                                    | 81.11       |
| BNNF [42]                                   | 80.96       |
| AIRS[43]                                    | 84.50       |
| Hybrid neural network[44]                   | 87.40       |
| Neural networks ensemble [45]               | 89.01       |
| Mean selection method [46]                  | 81.75       |
| Half selection method [46]                  | 83.44       |
| Neural network for threshold selection [46] | 84.46       |
| PSO + ELM [47]                              | 89.47       |
| SRLPSO + ELM [47]                           | 91.33       |
| (existence)C-ANFIS                          | 88.52       |
| (num)C-ANFIS                                | 73.77       |

ム・フレームワークで、探索空間上のサンプリング点の情報ベクトルを 1 つの粒子と見立ててばら撒いた粒子群を用いて大域的に解探索を行う手法である [48]。PSO は速度ベクトルの更新によって複数の粒子を動かし、最も結果がよかった位置を記憶しながら最適解を探す。粒子の位置はランダムに配置されるので、学習結果は配置する粒子の数と確率で左右されるが、複数の局所解がある場合、最小でない局所解 (局所的最適解) に陥ることなく最適解を探すことが可能である。本論文での ANFIS の学習法は勾配降下法であったが、勾配降下法は Greedy 法により、現段階において最も目的関数 (損失関数) を最小に減らせると予想する方向に降下するという性質があるため、複数の局所解がある関数の場合、降下先が局所的最適解に陥ることがある。C-ANFIS は局所探索に優れており、PSO は広範囲の最適化問題に適している。今回は両者の違いが結果に表れたと考えられる。

学習方法の改善策として、ANFIS の学習に PSO を使用した PSO-ANFIS で C-ANFIS を学習する。Moayed の研究文献 [49] によれば、PSO-ANFIS 技術のモデル手順は、粒子のセットの初期化から始まり、粒子のランダム選択を続け、最後に各粒子の位置を設定する。局所的最適解を避けるために、まずは通常通り ANFIS の学習をした後、PSO によって MF の数とパラメータが更新される。PSO によって MF の数が減り、容量の削減も達成している。

#### 4.4. まとめ

クラス変数が existence の場合、C-ANFIS は他の機械学習と比較して上から 4 番目の性能を示したが、SRLPSO+ELM との Accuracy の差は 2.81[%] であり、さらなる精度向上が必要である。また、特徴量選択方法による精度への影響や、心臓病データセットにおける ANFIS との性能差は不明確であった。C-ANFIS の学習方法およびデータセットの作成方法には見直しが必要であり、FPGA への実装は見送った。

## 第5章 結論

本研究では、4 入力 3MF の ANFIS を FPGA に実装し、心臓病データセットを用いた C-ANFIS の性能評価を行うことで、C-ANFIS を実装した FPGA の医療 IoT デバイスとしての可能性を検討した。

まず、ANFIS の基本構造とその学習方法を整理し、MATLAB を用いた学習モデルを FPGA へ適用する手法を確立した。次に、FPGA 上で ANFIS を動作させ、処理速度、精度、消費電力の観点から CPU との比較を実施した。その結果、いくつかの知見が得られた。

処理精度の評価では、FPGA と CPU の分類精度はほぼ同等であった。アヤメデータセットでは、FPGA と CPU の Accuracy が 100[%] と高く、fp16 での設計による精度低下の影響は限定的であった。一方、天秤データセットでは、ANFIS の Accuracy は約 88[%] であり、アヤメデータセットと比較すると精度が低いが、同じデータセットで学習した既存の機械学習手法と同等の精度を示した。2つのデータセットのクラス変数に対する相関係数を算出し比較した結果、クラス変数に対する相関係数が弱い入力項目が多いデータセットを学習した ANFIS は、識別精度が低くなると考察した。

処理時間の比較では、FPGA の処理速度は ARM Cortex A9 よりも高速であり、Intel Core i7 よりも遅いが、クロック周波数を考慮すると十分な性能を発揮した。さらに、FPGA はデータセットによる処理時間のばらつきが少なく、安定した処理が可能であることが確認された。この特性は、リアルタイム性が求められる医療 IoT デバイスにおいて重要である。

消費電力の評価では、FPGA の消費電力が 1.4W 程度と低く、Intel Core i7 (12W~28W) に比べて約 10 分の 1 の消費電力で動作可能であることが分かった。これは、低消費電力が求められる IoT エッジデバイスとして、FPGA が有望であることを示唆している。

心臓病データセットを用いた C-ANFIS の性能評価では、クラス変数の前処理や特徴量選択を行い、作成したデータセットの比較検証を行った。各データセットによって性能が高いクラス変数は異なり、C-ANFIS と ANFIS のどちらが性能が優れているかを明確に検証できるデータセットは存在しなかった。このため、特徴量選択の詳細な条件付けやテストデータの作成方法については、さらなる工夫が必要であるという課題が残された。

心臓病データセットは、項目数を削減したデータセット 1~5 と比較して各評価項目の性能が高く、データセット 1~5 を学習した ANFIS よりも優れていた。このことから、入力項目を削減したデータセットを ANFIS で学習するよりも、全ての入力項目を含むデータセットを C-ANFIS で学習する方が、精度の高い分類が可能であることが明らかになった。

作成した C-ANFIS を他機械学習と比較した結果、Neural networks ensemble、PSO + ELM、Proposed SRLPSO + ELM に次いで 4 番目の性能であった。1 番目の性能である Proposed SRLPSO + ELM との Accuracy の差は 2.81[%] であり、C-ANFIS のさらなる精度の向上が

求められる。PSO + ELM、Proposed SRLPSO + ELM と C-ANFIS との性能の違いは学習方法によると考えられる。C-ANFIS は局所探索に優れており、PSO は広範囲の最適化問題に適している。両者の違いが結果に表れたと考えられる。したがって学習方法の改善が必須と考える。

C-ANFIS の学習方法および学習用データセットの作成方法には見直しが必要であり、FPGA への実装は見送った。

心臓病の診断を行うエッジデバイスの作成に関しては、これまでの研究文献ではほとんど取り扱われておらず、したがってその有用性を検証するための十分なデータが存在しない。このため、本研究では新たにエッジデバイスを作成しても、その性能や効果を客観的に評価することが難しいという課題がある。さらに本論文では、心臓病の診断に使用する C-ANFIS を FPGA に実装する段階には至っていない。今後の研究課題として、4 入力 3MF の ANFIS の実装を応用することで C-ANFIS を FPGA に実装し、その性能を評価することが挙げられる。医療用エッジデバイスの実装が成功すれば、心臓病の診断精度の向上や診断の迅速化が期待できるが、そのためにはさらなる技術開発と検証が必要である。

## 参考文献

- [1] Somayya Madakam, Siddharth Tripathi, Ramya Ramaswamy: "Internet of Things (IoT): A literature review", *Journal of Computer and Communications*, Vol. 3, No. 5, pp. 164–173, 2015.
- [2] Junaid Iqbal Khan, Furqan Ali Farman Ullah Jamshid Bacha Sungchang Lee, Jebran Khan: "Artificial Intelligence and Internet of Things (AI-IoT) Technologies in Response to COVID-19 Pandemic: A Systematic Review", *IEEE Access*, Vol. 10, pp. 62613–62660, 2022.
- [3] Ram, S Dhanalakshmi S Vidya P Meenakshi V Vijay hari , H Vishal: "Regulation of water in agriculture field using Internet Of Things", *2015 IEEE Technological Innovation in ICT for Agriculture and Rural Development (TIAR)*, pp. 112–115. IEEE, 2015.
- [4] Ayush Kapoor, Sushila Shidnal Akshay Mehra, Suchetha I Bhat: "Implementation of IoT (Internet of Things) and Image processing in smart agriculture", *2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*, pp. 21–26. IEEE, 2016.
- [5] Jihen Amara, Alsayed Algergawy, Bassem Bouaziz: "A deep learning-based approach for banana leaf diseases classification", 2017.
- [6] 山下芳範: "病院における IoT 活用と環境", *IEICE Conferences Archives*. The Institute of Electronics, Information and Communication Engineers, 2018.
- [7] 中川肇鹿妻洋之 藤沢悟 定仲信行 保坂雅樹 酒谷薫: "健康医療介護分野における IoT デバイス活用の現状と将来像", *医療情報学*, Vol. 39, No. 5, pp. 239–246, 2020.
- [8] Alexander Magyari, Yuhua Chen: "Review of state-of-the-art FPGA applications in IoT Networks", *Sensors*, Vol. 22, No. 19, p. 7496, 2022.
- [9] Mohammed Elnawawy, Ahmad Al Nabulsi Abdul-Rahman Al-Ali Assim Sagahyroon, Abid Farhan: "Role of FPGA in internet of things applications", *2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 1–6. IEEE, 2019.
- [10] Murad Qasaimeh, Jack Lo Kees Vissers-Joseph Zambreno-Phillip H Jones, Kristof Denolf: "Comparing energy efficiency of CPU, GPU and FPGA implementations for vision kernels", *2019 IEEE international conference on embedded software and systems (ICESSE)*, pp. 1–8. IEEE, 2019.

- [11] Muzaffar Rao, Thomas Newe, Joseph Coleman: "An FPGA based reconfigurable IPSec ESP core suitable for IoT applications", *2016 10th International Conference on Sensing Technology (ICST)*, pp. 1–5. IEEE, 2016.
- [12] 厚生労働省: "令和5年(2023)人口動態統計(確定数)の概況", <https://www.mhlw.go.jp/>, 2023.
- [13] 厚生労働省: "我が国の人口について", [https://www.mhlw.go.jp/stf/newpage\\_21481.html](https://www.mhlw.go.jp/stf/newpage_21481.html).
- [14] J.-S.R.Jang: "ANFIS: adaptive-network-based fuzzy inference system", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, No. 3, pp. 665–685, 1993.
- [15] Takagi Tomohiro, Sugeno Michio: "Fuzzy identification of systems and its applications to modeling and control", *IEEE transactions on systems, man, and cybernetics*, No. 1, pp. 116–132, 1985.
- [16] N.Khaehintung, W.Kurutach, P.Sirisuk: "A novel ANFIS controller for maximum power point tracking in photovoltaic systems", *The Fifth International Conference on Power Electronics and Drive Systems, 2003. PEDS 2003.*, Vol. 2, pp. 833–836 Vol.2, 2003.
- [17] Chiung-Wei Huang, Jun-Tin Zhou Cheng-Yuan Chang, Shing-Tai Pan: "Enhanced temperature control method using ANFIS with FPGA", *The Scientific World Journal*, Vol. 2014, No. 1, p. 239261, 2014.
- [18] S Agatonovic-Kustrin, Rosemary Beresford: "Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research", *Journal of pharmaceutical and biomedical analysis*, Vol. 22, No. 5, pp. 717–727, 2000.
- [19] Guerra, et al., Maria IS: "Survey on adaptative neural fuzzy inference system (ANFIS) architecture applied to photovoltaic systems", *Energy Systems*, Vol. 15, No. 2, pp. 505–541, 2024.
- [20] Hussain Sajid, Al Alili Ali: "Soft computing approach for solar radiation prediction over Abu Dhabi, UAE: A comparative analysis", *2015 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, pp. 1–6, 2015.
- [21] Meisam Babanezhad, Ali Taghvaei Nakhjiri-Azam Marjani Saeed Shirazian, Armin Masoumian: "Influence of number of membership functions on prediction of membrane systems using adaptive network based fuzzy inference system (ANFIS)", *Scientific Reports*, Vol. 10, No. 1, p. 16110, 2020.
- [22] Aynur Yonar, Harun Yonar: "Modeling air pollution by integrating ANFIS and metaheuristic algorithms", *Modeling Earth Systems and Environment*, Vol. 9, No. 2, pp. 1621–1631, 2023.
- [23] Saeed Nosratabadi, Zoltan Lakner Csaba Mako Amir Mosavi, Sina Ardabili: "Prediction of food production using machine learning algorithms of multilayer perceptron and ANFIS", *Agriculture*, Vol. 11, No. 5, p. 408, 2021.

- [24] Maryam Mokhtarzad, Nima Jamshidi Vanjani Alireza Arabasadi, Farzad Eskandari: "Drought forecasting by ANN, ANFIS, and SVM and comparison of the models", *Environmental earth sciences*, Vol. 76, pp. 1–10, 2017.
- [25] Kandarpa Kumar Sarma, Vikash Mishra Samadrita Bhuiya Dmitrii Kaplun, Kunal Kingkar Das: "Learning Aided System for Agriculture Monitoring Designed Using Image Processing and IoT-CNN", *IEEE Access*, Vol. 10, pp. 41525–41536, 2022.
- [26] Yilmaz.I, Kaynar.Oguz: "Multiple regression, ANN (RBF, MLP) and ANFIS models for prediction of swell potential of clayey soils", *Expert systems with applications*, Vol. 38, No. 5, pp. 5958–5966, 2011.
- [27] Danial Jahed Armaghani, Panagiotis G Asteris: "A comparative study of ANN and ANFIS models for the prediction of cement-based mortar materials compressive strength", *Neural Computing and Applications*, Vol. 33, No. 9, pp. 4501–4532, 2021.
- [28] SO Sada, SC Ikpeseni: "Evaluation of ANN and ANFIS modeling ability in the prediction of AISI 1050 steel machining performance", *Heliyon*, Vol. 7, No. 2, 2021.
- [29] Ahmed El-Shafie, Aboelmagd Noureldin, Mahmoud Reda Taha: "A neuro-fuzzy model for inflow forecasting of the Nile river at Aswan high dam", *Water resources management*, Vol. 21, pp. 533–556, 2007.
- [30] Sajjad M Vatanchi, Mahmoud F Maghrebi Rouzbeh Shad, Hossein Etemadfar: "A comparative study on forecasting of long-term daily streamflow using ANN, ANFIS, BiLSTM and CNN-GRU-LSTM", *Water Resources Management*, Vol. 37, No. 12, pp. 4769–4785, 2023.
- [31] Satia Nur Maharani, Makaryanawati Makaryanawati Mohammad Mahbubi Ali, Bambang Sugeng: "Bank Soundness Level Prediction: ANFIS vs Deep Learning", *Journal of Applied Data Sciences*, Vol. 4, No. 3, pp. 175–189, 2023.
- [32] Nabeel Shirazi, Peter Athanas, Al Walters: "Quantitative analysis of floating point arithmetic on FPGA based custom computing machines", *Proceedings IEEE Symposium on FPGAs for Custom Computing Machines*, pp. 155–162. IEEE, 1995.
- [33] HyunJin Kim, Kang-Il Choi: "A Pipelined Non-Deterministic Finite Automaton-Based String Matching Scheme Using Merged State Transitions in an FPGA", *PloS one*, Vol. 11, No. 10, p. e0163535, 2016.
- [34] Fisher, R. A.: "Iris". UCI Machine Learning Repository, 1988.
- [35] Siegler, R.: "Balance Scale". UCI Machine Learning Repository, 1994.
- [36] Uday Shankar Shanthamallu, Huan Song Andreas Spanias, Jayaraman J Thiagarajan: "Gramme: Semisupervised learning using multilayered graph attention models", *IEEE transactions on neural networks and learning systems*, Vol. 31, No. 10, pp. 3977–3988, 2019.

- [37] Namal Rathnayake, Yukinobu Hoshino, Tuan Linh Dang: "A novel optimization algorithm: Cascaded adaptive neuro-fuzzy inference system", *International Journal of Fuzzy Systems*, Vol. 23, No. 7, pp. 1955–1971, 2021.
- [38] Marwah Mohammad Almasri, Abrar M Alajlan: "A novel-cascaded ANFIS-based deep reinforcement learning for the detection of attack in cloud IoT-based smart city applications", *Concurrency and Computation: Practice and Experience*, Vol. 35, No. 22, p. e7738, 2023.
- [39] 平良由香利大釜徳政 鈴木純恵 内海香子 佐藤佳子 和久紀子: "虚血性心疾患患者における自己管理行動に関する文献考察", 獨協医科大学看護学部紀要, Vol. 7, pp. 51–64, 2014.
- [40] Andras Janosi, Matthias Pfisterer Robert Detrano, William Steinbrunn: "Heart Disease". UCI Machine Learning Repository, 1989. DOI:.
- [41] Robert Detrano, Walter Steinbrunn Matthias Pfisterer Johann-Jakob Schmid Sarbjit Sandhu Kern H Guppy Stella Lee Victor Froelicher, Andras Janosi: "International application of a new probability algorithm for the diagnosis of coronary artery disease", *The American journal of cardiology*, Vol. 64, No. 5, pp. 304–310, 1989.
- [42] Cheung, Newton: "Machine learning techniques for medical analysis", *School of Information Technology and Electrical Engineering*, 2001.
- [43] K. PolatH. Kodaz S. Gunes: "A new classification method to diagnosis heart disease: supervised artificial immune system(AIRS)", *Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN '05)*, 2005.
- [44] H. Kahramanli, N. Allahverdi: "Design of a hybrid system for the diabetes and heart diseases", *Expert Systems with Applications*, vol. 35, no. 1-2, pp. 82–89, 2008.
- [45] Resul Das, Abdulkadir Sengur, Ibrahim Turkoglu: "Effective diagnosis of heart disease through neural networks ensembles.", *Expert Syst. Appl.* 36 (2009): 7675-7680., 2009.
- [46] Jaganathan Palanichamy, Ramasamy Kuppuchamy: "A threshold fuzzy entropy based feature selection for medical database classification.", *Computers in biology and medicine* 43 12: 2222-9, 2013.
- [47] C. V. Subbulakshmi, S. N. Deepa: "Medical Dataset Classification: A Machine Learning Paradigm Integrating Particle Swarm Optimization with Extreme Learning Machine Classifier.", *The Scientific World Journal* 2015, 2015.
- [48] Tuan Linh Dang, Yukinobu Hoshino: "Improved pso algorithm for training of neural network in co-design architecture", *International Journal of Computer Applications*, Vol. 975, p. 8887, 2019.

- [49] Hossein Moayedi, Mehdi, Raftari, Wan Amizah Wan Jusoh Ahmad Safuan A Rashid, Abolhasan Sharifi: "Optimization of ANFIS with GA and PSO estimating  $\alpha$  ratio in driven piles", *Engineering with computers*, Vol. 36, No. 1, pp. 227–238, 2020.