

珠算学習における計算結果の逐次分析による 苦手操作検出手法の提案

Proposal of a Method for Detecting Weak Operations by Sequential Analysis of Calculation Results in Abacus Learning

小嶋 泰造^{1*} 松田 裕貴¹
Taizo Kozaki¹ Yuki Matsuda¹

¹ 奈良先端科学技術大学院大学
¹ Nara Institute of Science and Technology

Abstract: 算盤学習者が珠算能力を向上させるためには、学習者の苦手操作や計算ミスを見つけ出し、指導することが必要である。しかしながら、この方法には指導者という人手が必要である。この現状から、学習者の操作をセンシングし、個々の状況に応じた学習支援を提供することで、算盤学習の効率を向上させることが可能だと考えた。著者らはこれまでに、市販の算盤に AR マーカを枠に貼り付け書画カメラによって撮影した俯瞰映像を分析することで、算盤の盤面を認識し、認識結果に基づき珠算学習支援に必要な情報をリアルタイムに卓上ディスプレイ上へと提示する手法を提案している。また、珠算学習にゲーム要素を取り入れ、楽しみながら繰り返し学習できるような仕組みを実現することで、苦手操作を効果的に克服するための珠算学習支援システムも提案している。しかしながら、苦手操作の検出および苦手克服のための問題生成に関する具体的な手法については提案されていない。本研究では、珠算学習者の苦手操作を検出する珠算学習支援手法の確立を目指している。本稿では、著者らがこれまでに提案してきたシステムを使用することを前提とする。算盤の盤面の認識結果とディスプレイに表示された計算問題の正誤判定を行い、誤答と判定した場合に認識結果の値と出題された計算問題の正解の値の差分を算出し、差分結果から学習者の苦手操作を検出する手法を提案する。

1 はじめに

計算能力を獲得するための方法の一つとして、現代でも活用されているのが算盤（そろばん）を用いた数値計算、すなわち「珠算」が存在する。算盤は、串で刺した珠を移動させることでその位置で数を表現し、四則演算を含む様々な計算を補助する道具である。物理的な珠の操作と視覚的な算盤上での数値表現を組み合わせることで、一般的な計算とは異なる方法で計算を実現する。近年では、直接的に算盤を使用する場面は減少しているが、珠算学習を通じて習得できる能力や効果が注目されている [1, 2, 3, 4, 5, 6]。算盤の学習方法として、算盤教室や学校教育が一般的である。具体的な指導方法として、指導者が学習者の珠操作を観察し、苦手操作や計算ミスを見つけ出すことや正しい操作方法を実演することで理解を促し、指導する方法がある。しかし、これらの方法は指導者に依存しており、

労力が必要とされる。また、珠算学習において、正確な珠操作を習得することが重要であるが、珠操作の種類は多様であり、学習者が苦手操作に直面することも多く存在する。苦手操作を克服するためには、長期的な繰り返し練習が必要であるが、苦手操作ばかりを練習することは学習者にとってストレスや嫌悪感を引き起こす可能性がある。

著者らはこれまでに、市販の算盤に AR マーカを枠に貼り付け書画カメラによって撮影した俯瞰映像を分析することで、算盤の盤面を認識し、認識結果に基づき珠算学習支援に必要な情報をリアルタイムに卓上ディスプレイ上へと提示する手法を提案している [7, 8, 9]。また、珠算学習にゲーム要素を取り入れ、楽しみながら繰り返し学習できるような仕組みを実現することで、苦手操作を効果的に克服するための珠算学習支援システムも提案している [10]。しかしながら、苦手操作の検出および苦手克服のための問題生成に関する具体的な手法については提案されていない。

本研究では、珠算学習者の苦手操作を検出する珠算

*連絡先：奈良先端科学技術大学院大学
〒 630-0192 奈良県生駒市高山町 8916 番地-5
E-mail: kozaki.taizo.kv7@is.naist.jp

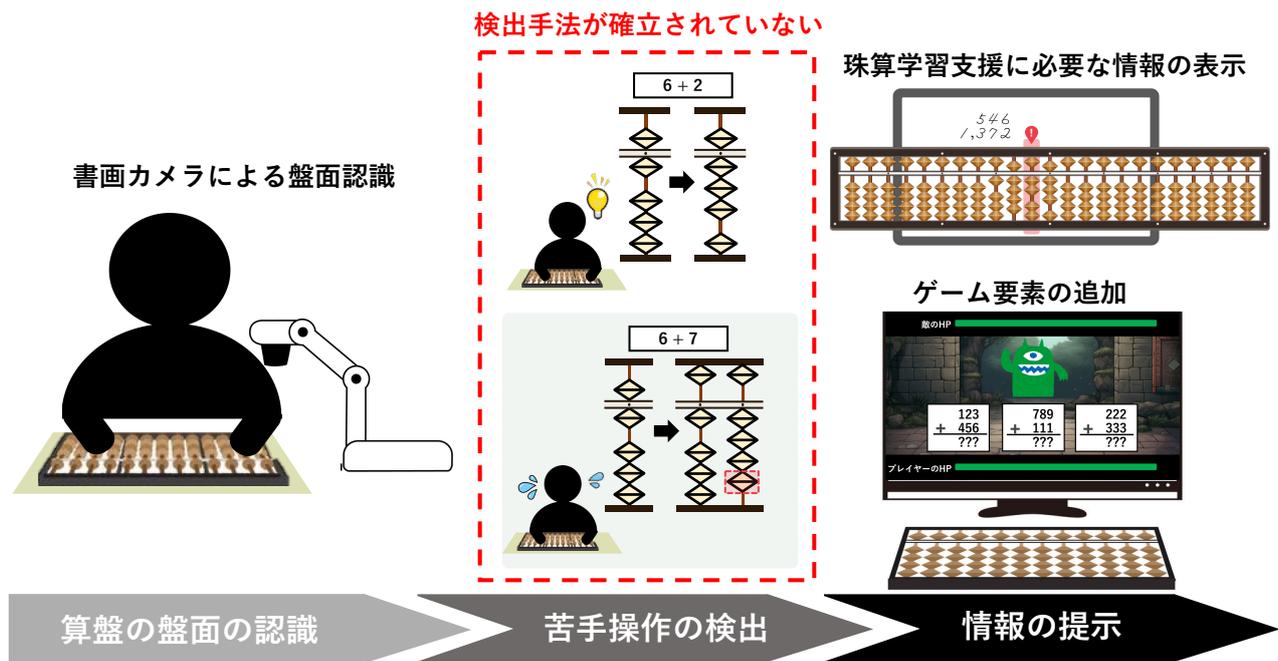


図 1: 珠算学習支援システムの全体像

学習支援手法の確立を目指している。本稿では、著者らがこれまでに提案してきたシステム [7, 8, 9, 10] を使用することを前提とする。算盤の盤面の認識結果とディスプレイに表示された計算問題の正誤判定を行い、誤答と判定した場合に認識結果の値と出題された計算問題の正解の値の差分を算出し、差分結果から学習者の苦手操作を検出する手法を提案する。

2 関連研究・サービス

本章では、珠算学習支援システムに関する研究・サービスについて述べるとともに、本研究の立ち位置を示す。

2.1 珠算学習支援に関する関連研究・サービス

北川らは、プロジェクタ、LED ライト、RGB カメラを使用して、リアルタイムで算盤の盤面推定および算盤上に計算支援情報を表示する珠算学習支援システムを提案している [11]。このシステムは、算盤学習者が実際の算盤を使用し、計算手順や指使いをプロジェクションマッピングにより視覚情報として提示することで学習を行う。

新川らは、珠算教育向けの Web 学習支援システムを提案しており、フラッシュ暗算や読上算、見取り算などの学習用ソフトを使用し、Learning Management System

(LMS) 上で成績管理が行える。また、個人成績や学習の進捗状況と組み合わせることによって、ソフトウェアが搭載された PC であれば、場所を問わずの学習が可能となる [12]。

スマートフォンやタブレット端末のディスプレイ上に算盤を再現し、珠算学習を支援する手法も存在する。齋藤らは、スマートフォンやタブレット端末を利用した珠算学習支援ツールを提案している [13]。端末の画面上で算盤を再現し、珠を指で弾く操作を実現している。また、端末内の加速度センサを使用し、端末を傾けることで御破算操作を行うことも可能となっている。

Digika 社は、「そろタッチ」というサービス [14] を提供しており、算盤の UI や操作方法を応用した暗算学習支援が可能となっている。このサービスでは、タブレット端末上で算盤に似たインターフェースを再現し、珠に対応するボタンを両手で操作することで計算を行う。

著者らはこれまでに、市販の算盤に AR マーカを枠に貼り付け、書画カメラによって撮影した俯瞰映像を分析することで、簡易に算盤の盤面を認識する手法を提案している [7, 8]。また、これらの手法を拡張し、盤面認識の結果に基づき珠算学習支援に必要な情報をリアルタイムに卓上ディスプレイ上へと提示する手法も提案している [9]。さらに、この手法 [9] をベースとし、苦手操作の抽出・問題への反映を行う手法およびゲーム要素を追加することで、算盤の苦手操作克服のための珠算学習支援手法も提案している [10]。このシステ

ムを使用し、珠算学習を楽しく繰り返し学習できるようにすることで、ストレスや心理的な負荷を感じることなく苦手操作の克服を目指している。

2.2 本研究の立ち位置

算盤学習者が珠算能力を向上させるためには、学習者の苦手操作や計算ミスを見つけ出し、指導することが必要である。しかしながら、この方法には指導者という人手が必要である。この現状から、学習者の操作をセンシングし、個々の状況に応じた学習支援を提供することで、算盤学習の効率を向上させることが可能だと考えた。

これらの問題を解決するために、著者らはこれまでに、市販の算盤に AR マーカを枠に貼り付け書画カメラによって撮影した俯瞰映像を分析することで、算盤の盤面を認識し、認識結果に基づき珠算学習支援に必要な情報をリアルタイムに卓上ディスプレイ上へと提示する手法を提案している [7, 8, 9]。また、珠算学習にゲーム要素を取り入れ、楽しみながら繰り返し学習できるような仕組みを実現することで、苦手操作を効果的に克服するための珠算学習支援システムも提案している [10]。しかしながら、苦手操作の検出および苦手克服のための問題生成に関する具体的な手法については提案されていない。

本稿では、著者らがこれまでに提案してきたシステム [7, 8, 9, 10] を用いて、システムが誤答と判定した場合に認識結果の値と出題された計算問題の正解の値の差分を算出し、差分結果から学習者の苦手操作を検出する手法を提案する。

3 算盤の計算方法と苦手操作の定義

算盤は、串で刺した珠を移動させその位置で数を表現することで、四則演算を含む様々な計算を補助することが可能なツールである。ここでは、算盤の基本的な知識についての概説と苦手操作の定義について述べる。

3.1 算盤での数表現

まず、算盤での数の表現方法について説明する。図2は算盤の部位を示している。各「桁」は10進数の1桁に対応し、5つの珠の配置によって0から9までの数を表現できる。「桁」は「梁」を挟んで上段と下段に分かれており、上段の1つの珠（「五珠」）は10進数の5を、下段の4つの珠（「一珠」）はそれぞれ10進数の1を表現するために使われる。具体的な数の表現例を図3に示す。上段と下段の珠の間には珠の高さの半分のスペースがあり、珠を上下に動かすことで各珠の

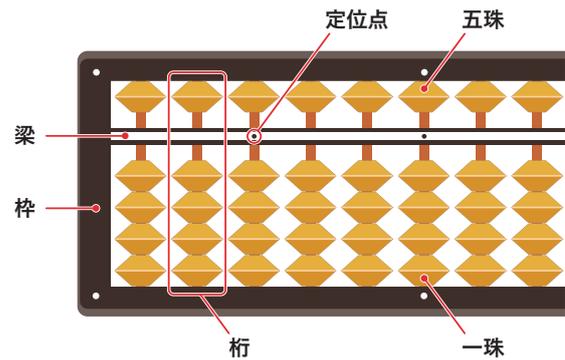


図2: 算盤の部位

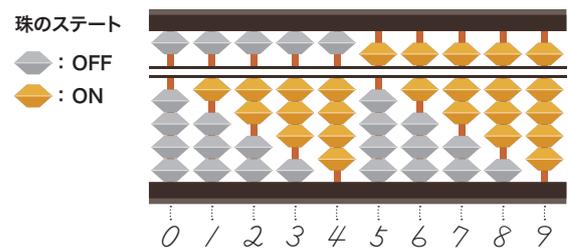


図3: 算盤での数表現

状態を表現する。上段では、珠が上がっている状態が OFF (0)、下がっている状態が ON (5) を示し、逆に下段では、珠が上がっている状態が ON (1)、下がっている状態が OFF (0) を示す。同じ桁の珠の総和 (ON の状態となっている珠の表す数値の合計) がその桁の数を示す。図2の「定位点」は、3桁ごとに梁に示されており、一の位を決める目印として使われる。一の位として決めた定位点より右側の桁は、小数を表現するために使用される。

3.2 算盤での計算方法

算盤を使用した計算は、主に5の補数と10の補数の組み合わせを利用する。具体的には、表1と表2に示す16つの操作が存在する。次では加算を例に挙げているが、減算も同様に、加算の逆の操作を行うことで表現できる。

操作1・2: 一珠のみ・五珠のみを操作する計算 計算の例を次に示す。操作1に該当する「 $1+2$ 」の計算では、最初に一珠が1つ上がっている状態 (1) から、さらに一珠を2つ上げる操作 (+2) を行うことで、3の状態を表現することができる。同様に、操作2に該当する「 $3+5$ 」の計算では、最初に一珠が3つ上がっている状態 (3) から、五珠を下げる操作 (+5) を行うことで、8の状態を表現することができる。

表 1: 加算操作の一覧

加算	一珠	五珠	繰り上げ
操作 1	+		
操作 2		+	
操作 3	+	+	
操作 4	-	+	
操作 5	-		+
操作 6		-	+
操作 7	-	-	+
操作 8	+	-	+

表 2: 減算操作の一覧

減算	一珠	五珠	繰り下げ
操作 1	-		
操作 2		-	
操作 3	-	-	
操作 4	+	-	
操作 5	+		-
操作 6		+	-
操作 7	+	+	-
操作 8	-	+	-

操作 3・4: 一珠・五珠を操作する計算 具体的には, 0~4 の数値から 5~9 の数値への状態変化 (またはその逆) が生じる操作に対応する. 計算の例を次に示す. 操作 3 に該当する「1+6」の計算では, 最初に一珠が 1 つ上がった状態 (1) に対して, (+6) すなわち, 五珠を下げる操作 (+5) とさらに一珠を 1 つ上げる操作 (+1) を加えることで, 計算結果の 7 を表現することができる. 一方で, 操作 4 に該当する「2+4」の計算では, 最初に一珠が 2 つ上がった状態 (2) に対し (+4) を加えるのだが, 一珠は一桁につき 4 つであることから一珠の組み合わせのみではこの計算は表現できない. そこで, 5 の補数の考え方を導入する. (+4) は「5-1」と表現できることから, 現在の入力値である 2 から 1 を減ずる操作, すなわち一珠を 1 つ下げる操作 (-1) と 5 を足す, すなわち五珠を下げる操作 (+5) を行うという操作を同時に行うことで, 「2+4」の計算結果である 6 の状態を表現することができる.

操作 5~8: 二桁を操作する計算 (繰り上げ・繰り下げ) 具体的には, 一桁で表現可能な 0~9 の数値から二桁で表現する 10 以上の数値への状態変化 (またはその逆), つまり繰り上がり・繰り下がりが生じる操作に対応する. 計算の例を次に示す. 操作 5 に該当する「3+8」の計算では, 最初に一珠が 3 つ珠が上がった状態 (3) に対し (+8) を加えるのだが, 一桁の範囲では表現できないため, 10 の補数の考え方を導入する. 8 は「10-2」と表現できることから, 現在の入力値である 3 から

2 を減ずる操作, すなわち一珠を 2 つ下げる操作 (-2) と 10 を足す, すなわち左隣の桁の一珠を 1 つ上げる操作 (+10) という操作を同時に行うことで, 「3+8」の計算結果である 11 の状態を表現することができる. 操作 6 に該当する「6+5」の計算では, 最初に一珠が 1 つ上がった状態と五珠が下がった状態である (6) に対し 5 を加えるのだが, 五珠はすでに下がっている状態に五珠を加算する操作 (+5), すなわち五珠を下げる操作は表現できない. そのため, 10 の補数の考え方を導入する. 5 は「10-5」と表現できることから, 現在の入力値である 5 から 5 を減ずる操作, すなわち五珠を上げる操作 (-5) と 10 を足す操作, すなわち左隣の桁の一珠を 1 つ上げる操作 (+10) 操作を同時に行うことで, 「6+5」の計算結果である 11 の状態を表現することができる. また, 操作 7 に該当する「8+3」の計算では, 最初に一珠が 3 つ珠が上がった状態 (3) と五珠が下がった状態に対し 3 を加えるのだが, 一桁の範囲では表現できないため, 10 の補数の考え方を導入する. 3 の 10 の補数「10-7」を考え, 1 の位から 7 を減じなければならぬので, 8 から 7 を減ずる. その後 10 を足す操作, すなわち左隣の桁の一珠を 1 つ上げる操作 (+10) を行うことで, 「8+3」の計算結果である 11 の状態を表現することができる. さらに, 10 の補数と 5 の補数を同時に考えなければならない場合もある. 操作 8 に該当する「6+8」の計算では, 8 は上述したとおり 10 の補数と考え, 1 の位から 2 を減じなければならないが, 6 から 2 を減ずるには 5 の補数「5-3」を考える必要がある. つまり, 「6+8」の計算結果である 14 を得るためには「3 を足す」「5 を減ずる」「10 を足す」という 3 つの操作を同時に行うことで「6+8」の計算結果である 14 の状態を表現することができる.

3.3 苦手操作の定義

珠算学習において, 正確な珠操作を習得することが重要であるが, 珠操作の種類は多様であり, 学習者が苦手操作に直面することも多く存在する. 本稿では, 独自で加算と減算の際の 6 パターンの苦手操作を定義し, 想定される具体例とともに説明する.

パターン 1: 加算操作の不足 は, 本来足すべき数字を足していない苦手操作である.

パターン 1 の苦手操作の例を図 4 に示す. 加算操作の操作 8 に該当する 6+8 の計算で想定する. 前述した通り, 6+8 の計算は「3 を足す」「5 を減ずる」「10 を足す」の 3 つの操作を同時に行う. しかし, 複数の操作を同時に行わなければならない場合, 操作の過程を忘れる可能性がある. そのため, 最初に操作する「3 を足す」という操作が行なわれない場合が存在する. その結果, 「5

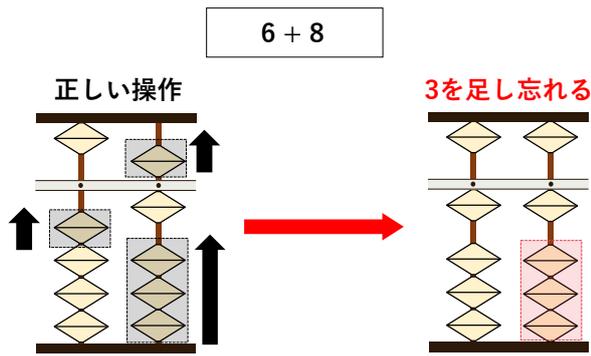


図 4: パターン 1 の加算操作の不足の例

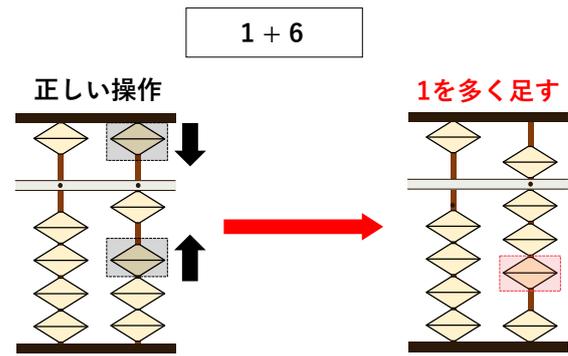


図 6: パターン 3 の加算量の錯誤の例

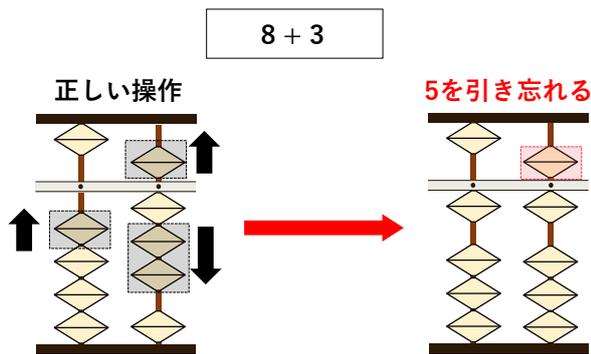


図 5: パターン 2 の減算操作の不足の例

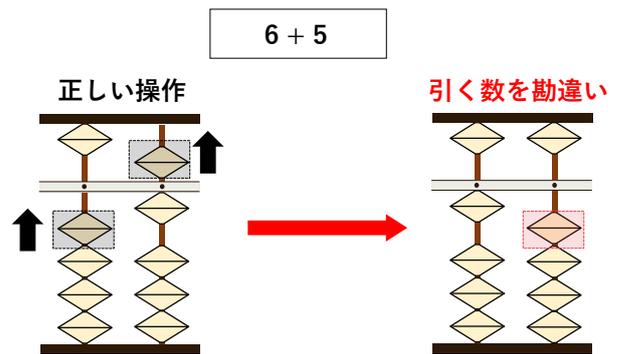


図 7: パターン 4 の減算量の錯誤の例

を減ずる」「10を足す」の操作のみが行なわれる場合が考えられる。

パターン 2: 減算操作の不足 は、本来引くべき数字を引いていない苦手操作である。

パターン 2 の苦手操作の例を図 5 に示す。加算操作の操作 7 に該当する $8+3$ の計算で想定する。前述した通り、 $8+3$ の計算は「7を減ずる」「10を足す」の二つの操作を同時に行う。しかし、複数の操作を同時に行わなければならない、操作の過程を忘れる可能性がある。そのため、最初に操作する「7を減ずる」という操作の中で「2を減ずる」は行なわれるが「5を減ずる」が行なわれない場合が存在する。その結果、「2を減ずる」「10を足す」の操作のみが行なわれる場合が考えられる。

パターン 3: 加算量の錯誤 は、本来足すべき数字より多い数または少ない数を足す苦手操作である。

パターン 3 の苦手操作の例を図 6 に示す。加算操作の操作 3 に該当する $1+6$ の計算で想定する。前述した通り、 $1+6$ の計算は「1を足す」「5を足す」の二つの操作を同時に行う。しかし、 $1+6$ の計算結果である 7 を暗算で計算してしまい、計算結果を直接入力してしまう可能性がある。そのため、「1を足す」という操作が「2を足す」という操作になり、その結果、計算結果が

8 になり、本来足すべき数字より多い数を足す場合が考えられる。

パターン 4: 減算量の錯誤 は、本来引くべき数字より多い数または少ない数を引く操作である。

パターン 4 の苦手操作の例を図 7 に示す。加算操作の操作 6 に該当する $6+5$ の計算で想定する。前述した通り、 $6+5$ の計算は「5を減ずる」「10を足す」の二つの操作を同時に行う。しかし、6 という数字が先に目に入り、勘違いから 6 から「5を減ずる」操作を誤って「6を減ずる」操作をしてしまい、誤った減算をしてしまう可能性がある。そのため、「5を引く」という操作が「6を引く」という操作になり、その結果、計算結果が 10 になり、本来引くべき数字より多い数を引く場合が考えられる。

パターン 5: 五珠・十珠の錯誤 は、本来 5 を足すまたは引く操作を誤って 10 を足す、または、引いてしまう操作及び 10 を足すまたは引く操作を誤って 5 を足すまたは引いてしまう苦手操作である。

パターン 5 の苦手操作の例を図 8 に示す。加算操作の操作 2 に該当する $3+5$ の計算で想定する。前述した通り、 $3+5$ の計算は五珠を下げる操作 (+5) を行う。しかし、五珠を下げる操作 (+5) を誤って同じ一つの珠の

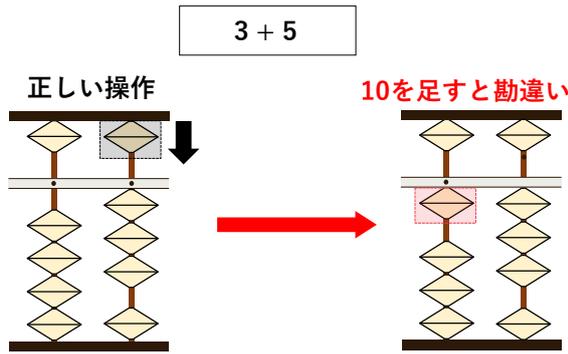


図 8: パターン 5 の五珠・十珠の錯誤の例

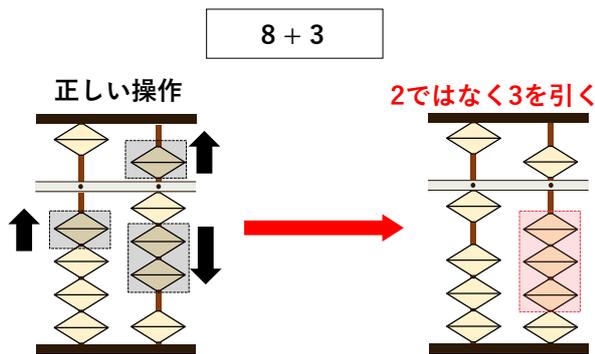


図 9: パターン 6 の加減操作の反転の例

みを操作する 10 を足す操作と勘違いしてしまう可能性がある。そのため、「5 を足す」という操作が「10 を足す」という操作になり、その結果、計算結果が 13 になり、本来 5 を足す操作を誤って 10 足してしまう場合が考えられる。

パターン 6: 加減操作の反転 は、本来足すべき数字を誤って引いてしまうまたは引くべき数字を誤って足してしまう操作である。

パターン 6 の苦手操作の例を図 9 に示す。加算操作の操作 7 に該当する $8+3$ の計算で想定する。前述した通り、 $8+3$ の計算は「7 を減ずる」「10 を足す」の二つの操作を同時に行う。しかし、10 の補数を勘違いし、そのまま「3 を減ずる」操作をしてしまう可能性がある。そのため、最初に操作する「7 を減ずる」操作を「3 を減ずる」と操作してしまう。その結果、「3 を減ずる」「10 を足す」の操作が行なわれ、本来足すべき数字を誤って引いてしまう場合が考えられる。

4 計算結果の逐次分析による苦手操作検出手法

4.1 提案の概要

本研究では、珠算学習者の苦手操作を検出する珠算学習支援手法の確立を目指している。本稿では、著者がこれまでに提案してきたシステム [7, 8, 9, 10] を使用することを前提とする。算盤の盤面の認識結果とディスプレイに表示された計算問題の正誤判定を行い、誤答と判定した場合に認識結果の値と出題された計算問題の正解の値の差分を算出し、差分結果から学習者の苦手操作を検出する手法を提案する。苦手操作は、誤操作の蓄積から推定する。同じ誤操作を複数回繰り返すことは、本来すべき操作を理解していないことや、忘れていている可能性が挙げられる。そのため、最初に誤操作を含んでしまった計算の認識結果の値と計算問題の正解の値の差分結果から前述した 6 パターンの苦手操作に分類し、一定回数の誤操作が蓄積された場合にその操作を苦手操作と推定する。次に誤操作の分類方法及び苦手操作の推定方法について詳細に説明する。

4.2 誤操作の分類

提案システムを使用し、誤操作により誤答と判定された場合に、算盤の盤面の認識結果の値と出題された計算問題の正解の値の差分を算出し、前述した 6 パターンの苦手操作に分類する。算盤で行う全ての操作は、一桁の計算を組み合わせた操作であり、表 1 と表 2 の操作で網羅できる。全ての操作に該当する 6 パターンの苦手操作の組み合わせを表 3 に示す。また、各操作の中に含まれる全ての一桁の計算に対して、6 パターンの苦手操作を行った際の計算結果の値と正解の値の差分結果はあらかじめシステム内にデータとして登録してあるものとする。次に差分結果による誤操作の分類方法について説明する。

基本的な誤操作の分類方法 前述した $6+8$ の計算で誤操作を分類する方法を例に示す。 $6+8$ は操作 8 に該当し、操作 8 は表 3 に示した通り、想定される 6 パターンの苦手操作を全て含む。例えばパターン 1 で考えると、計算中に含まれる「3 を足す」操作が行われていないことになる。つまり、認識結果の値が 11 になり、出題された計算問題の正解の値が 14 なので差分が -3 になる。操作 8 の中の $6+8$ の計算で差分が -3 になるものは苦手操作のパターン 1 になるとあらかじめデータとして登録してあり、分類することができる。他にパターン 3 で考えると、 $6+8$ の計算で、「3 を足す」操作を本来足すべき数字より少ない「2 を足す」操作になると、

表 3: 加算と減算の各操作に該当する苦手操作のパターン

	パターン 1 加算操作の不足	パターン 2 減算操作の不足	パターン 3 加算量の錯誤	パターン 4 減算量の錯誤	パターン 5 五珠・十珠の錯誤	パターン 6 加減操作の反転
操作 1	○	○	○	○		○
操作 2					○	
操作 3	○	○	○	○	○	
操作 4	○	○	○	○	○	○
操作 5	○	○	○	○	○	○
操作 6	○	○	○	○	○	○
操作 7	○	○	○	○	○	○
操作 8	○	○	○	○	○	○

認識結果の値が 13 になり、差分が -1 になる。差分が -1 になるものはこの場合、苦手操作のパターン 3 になるとあらかじめデータとして登録してあり、分類することができる。このように、一桁の計算に対して、誤操作を行った際の差分結果は 6 パターンの苦手操作ごとに値が違い、分類することが可能である。

二桁以上の計算における誤操作の分類方法 二桁以上の計算は、繰り上がりや繰り下がりの操作が他の桁に影響する。そのため、各操作の中に含まれる全ての一桁の計算に対して、6 パターンの苦手操作を行った際の差分結果以外の値になってしまうことがある。ここでは二桁+二桁の計算を例に挙げて二桁以上の計算における誤操作の分類方法について説明する。具体例として、 $12+48$ の計算を例に挙げる。まず十の位の計算結果から順番に差分を計算する。正解の値の十の位は 6 になるが、計算問題の十の位である $1+4$ の計算結果は 5 となる。そのため、正解の値である 6 の結果は「1 を減ずる」操作を忘れて「5 を足す」操作のみを行うパターン 2 に該当することになる。これでは間違った誤操作の分類を行うことになるので、まず初めに計算問題から一の位に繰り上がりがあるのかを確認し、繰り上がり操作があった場合は、繰り上がり操作があるというフラグを立てる。このフラグを確認した場合は、十の位は $2+4$ の計算の考える。その後は前述した基本的な誤操作の分類方法を適用する。次に、一の位の計算結果の差分を計算する。一の位は他の桁が影響しないので、前述した基本的な誤操作の分類方法を適用する。しかし、繰り上がり操作が正しく行われているか判断することができない。その際は、起こりうる苦手操作の該当するもの全てに分類する。また、今回は二桁+二桁の計算を例に示したが、二桁以上の計算でも同様に同じ方法を適用することで、誤操作を分類することが可能である。

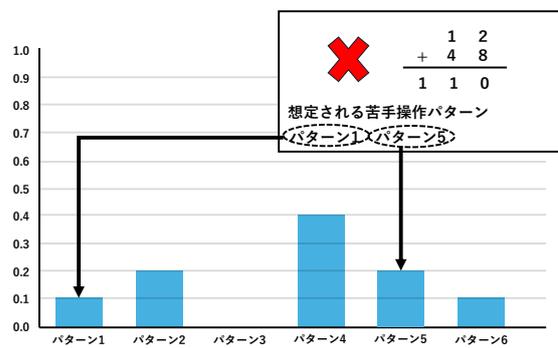


図 10: 苦手操作の推定方法の具体的なイメージ図

4.3 苦手操作の推定

提案システムを使用し、誤答と判定されるたびに誤操作はそれぞれのパターンに分類が行われ、データとして蓄積されていく。蓄積された分類結果から一定回数を超えたものや割合が大きいものを苦手操作と推定する。苦手操作の推定方法の具体的なイメージを図 10 に示す。この場合だと、 $12+48$ の計算結果が 110 となっており、誤答である。計算結果が 110 になる誤操作から想定される苦手操作パターンはパターン 1 とパターン 5 であり、それぞれに分類し、全体の分類結果を割合として考える。複数回にわたって誤操作の分類を続けると、それぞれのパターンの割合が変化し、特に回数が多い苦手操作パターンがあると、その割合は徐々に大きくなっていく。このようにして、割合が大きいものを苦手操作として推定することができる。

5 おわりに

本研究では、珠算学習者の苦手操作を検出する珠算学習支援手法の確立を目指している。本稿では、著者らがこれまでに提案してきたシステムを使用することを前提とし、認識結果の値と出題された計算問題の正

解の値の差分結果から学習者の苦手操作を検出する手法を提案した。今後は、提案した手法を組み込んだシステムの実現を目指す。また、実際の算盤学習者を対象に、短期的・中長期的な実験を行い、システムの実用性について評価を行う予定である。

謝辞

本研究の一部は、KDDI 財団調査研究助成（2021 年度）の助成を受けて行われたものです。

参考文献

- [1] Shizuko Amaiwa and Giyoo Hatano. Effects of abacus learning on 3rd-graders' performance in paper-and-pencil tests of calculation. *Japanese Psychological Research*, Vol. 31, pp. 161–168, 1989.
- [2] Shizuko Amaiwa. The Effects of Abacus Learning on Solving Arithmetic Problems: A Comparative Study of Elementary / Junior High School Students at Upper Level and Inexperienced Students. *Journal of the Faculty of Education, Shinshu University*, Vol. 96, pp. 145–156, 1999.
- [3] Chunjie Wang, Tianyong Xu, Fengji Geng, Yuzheng Hu, Yunqi Wang, Huafeng Liu, and Feiyan Chen. Training on Abacus-Based Mental Calculation Enhances Visuospatial Working Memory in Children. *Journal of Neuroscience*, Vol. 39, No. 33, pp. 6439–6448, 2019.
- [4] Chunjie Wang. A Review of the Effects of Abacus Training on Cognitive Functions and Neural Systems in Humans. *Frontiers in Neuroscience*, Vol. 14, No. 913, pp. 1–12, 2020.
- [5] Yujie Lu, Mengyi Li, Zhijun Cui, Li Wang, Yuwei Hu, and Xinlin Zhou. Transfer Effects of Abacus Training on Cognition. *Current Psychology*, Vol. 42, pp. 6271–6286, 2023.
- [6] Yuzheng Hu, Fengji Geng, Lixia Tao, Nantu Hu, Fenglei Du, Kuang Fu, and Feiyan Chen. Enhanced White Matter Tracts Integrity in Children With Abacus Training. *Human Brain Mapping*, Vol. 32, No. 1, pp. 10–21, 2011.
- [7] 松田裕貴. 書画カメラを用いた珠算行動センシング. 電子情報通信学会技術研究報告, センサネットワークとモバイルインテリジェンス研究会 (SeMI), 第 123 巻, pp. 70–75, 2023.
- [8] Yuki Matsuda. Abacus manipulation understanding by behavior sensing utilizing document camera as a sensor. In *The 5th International Conference on Activity and Behavior Computing (ABC '23)*, pp. 1–18, 2023.
- [9] 松田裕貴, 松田裕貴. 珠算学習支援のための盤面認識に基づくリアルタイム情報提示手法. マルチメディア, 分散, 協調とモバイル (DICOMO 2023) シンポジウム論文集, pp. 63–67, 2023.
- [10] 小寄泰造, 松田裕貴. 算盤の苦手操作克服のためのゲーム要素を用いた珠算学習支援手法. 第 31 回マルチメディア通信と分散処理ワークショップ (DPSWS'23), pp. 267–272, 2023.
- [11] 北川珠莉, 鈴木優. 珠の位置認識と操作手順の重畳表示を用いたそろばん学習支援システム. 情報処理学会インタラクシオン 2022, pp. 759–762, 2022.
- [12] 新川晃司, 川崎健志, 澤田一樹, 二石芳裕, 笥宗徳, 渡邊一衛. そろばん教育における Web 学習支援システムの開発. 成蹊大学理工学研究報告, 第 48 巻, pp. 75–79, 2011.
- [13] 齋藤謙太, 佐々木整, 水野一徳. 携帯電話を利用した学習支援ツールの開発. 情報科学技術フォーラム講演論文集, 第 8 巻, pp. 653–654, 2009.
- [14] 株式会社 Digika. そろタッチ. <https://www.sorotouch.jp/>. (accessed 2023-04-10).