修 士 論 文 の 和 文 要 旨

研究科・専攻	大学院　情報理工学研究科　　情報・ネットワーク工学専攻　博士前期課程		
氏　　名	赤坂尚紀	学籍番号	2131001
論 文 題 目	決定木を用いた 2 段階適応型テストの提案		

要　　旨

適応型テストは，受検者の能力値を逐次的に推定し，その能力値に応じて情報量が最大の項目を出題する Computer Based Testing の出題形式である．その利点として，能力値の推定精度を減少させずに出題項目数や受験時間を短縮できる．適応型テストでは，能力推定の漸近分散がフィッシャー情報量の逆数の値に収束することが知られているため，能力推定値に対してフィッシャー情報量が最大の項目を選択する．しかし，テスト開始直後には能力推定値と真の能力値が乖離しているため，最適でない項目が選択される傾向がある．この問題を解決するためにテスト開始直後の能力推定誤差を考慮した項目選択基準が提案されている．しかし，その項目選択基準の多くは，計算に数値積分が必要であるため，極めて計算コストが高く実用化が困難とされている．この問題を解決するために，受検者の全回答パターンに対して決定木を事前に構築し，その決定木を用いて項目を選択する手法が提案されている．しかし，この手法は，出題項目数の増加に伴い，決定木の分枝数が指数的に増加することから，構成可能な決定木の大きさが限定されてしまう問題がある．この問題を解決するために，本論文では，決定木を用いた 2 段階適応型テストを提案する．本手法は，事前に予測効率の高い項目選択基準を用いて構成可能な最大サイズの決定木を構築する．テスト時は，テスト前半に決定木に基づき項目選択し，受検者の能力値が収束し始めるテスト後半にフィッシャー情報量が最大の項目を選択する．第 1 段階では，極めて計算コストが高い項目選択基準を用いることによって能力推定誤差の影響を小さくできると期待できる．第 2 段階では，能力推定値は真の能力値に接近しているため，フィッシャー情報量を用いて項目選択することで高い能力推定精度が期待できる．本論では，シミュレーション実験と実データを用いた実験により決定木を用いた 2 段階適応型テストの有効性を示す．

令和４年度 修士論文

決定木を用いた２段階適応型テストの提案

電気通信大学大学院 情報理工学研究科
情報・ネットワーク工学専攻 情報数理工学プログラム

学籍番号 2131001

赤坂 尚紀

主任指導教員 植野 真臣 教授
指導教員 宇都 雅輝 准教授

2023 年 1 月 28 日

目次

1　まえがき

近年，e テスティングの実用化が進んでいる [1–3]．e テスティングは，異なる項目から構成されたテストの結果を同一尺度で評価でき，テストの結果が受検者に大きな影響を及ぼすハイ・ステークスなテストで導入されつつある．e テスティングの技術の一つとして適応型テスト（CAT: Computerized Adaptive Testing）と呼ばれるテスト出題方式が知られている [4]．適応型テストは，受検者の能力値を逐次的に推定し，その能力値に応じて情報量が最大の項目を出題する．これにより，能力値の推定精度を減少させずに出題項目数や受験時間を短縮できる．項目反応理論において，能力推定の漸近分散がフィッシャー情報量の逆数の値に収束することが知られている [26]．そのため，適応型テストでは一般的に能力推定値に対してフィッシャー情報量が最大の項目を選択する．しかし，テスト開始直後には能力推定値と真の能力値が乖離しているため，最適でない項目が選択される傾向がある．これは希薄化パラドックス（attenuation paradox）として古くから知られている [27]．この問題を解決するためにいくつかの情報量が提案されている．

Chang と Ying は，Kullback-Leibler 情報量に基づいた項目選択基準（以下，KL と呼ぶ）を提案している [7]．また，Veerkamp と Berger は，フィシャー情報量を尤度関数で重みづけ，その積分平均が最も高い項目を選択する手法を提案している [8]．これは，尤度重み付き情報量基準（likelihood weighted information criterion，以下 LWI と呼ぶ）と呼ばれている．さらに，van der Linden らは，能力値の事後分布でフィシャー情報量を重みづける項目選択手法（Maximum posterior-weighted information criterion，以下 MPWI と呼ぶ）を提案している [10]．また，能力値についての事後分布の分散を最小にする MEPV(Minimum Expected Posterior Variance)(van der Linden and Pashley, 2009) と呼ばれる項目選択基準が提案されている [21]．これらは，前述の問題を解消し，受検者の能力推定の信頼性を向上できた．しかし，こ

れらの手法は，能力パラメータ上で数値積分が必要であるため，極めて高い計算コストが要求される．適応型テストを運用するには，即時に項目を選択する必要があり，これらの手法が実用化されていない．

この問題を解決するために，Ueno and Songmuang(2010) [12] は，決定木に基づく項目選択手法を提案している．この手法では，受検者の全解答パターンに対して決定木を事前に構築し，その決定木を用いてアイテムバンクから項目を選択する．Ueno(2013) は，EVTI (Expected Value of Test Information) と呼ばれる項目選択基準を提案し，非常に計算コストが高い EVTI に基づき決定木を事前に構築することで，即時に項目を出題することを実現している．さらに，Ueno(2013) [13] らは，フィッシャー情報量 (MFI) と KL，LWI，EPWI，EVTI を比較し，EVTI が最も予測効率が高いことを報告している．しかし，決定木を用いた適応型テストは，テストの長さに伴い，分枝数が指数的に増加することから，時間 · 空間計算量が非常に大きくなりやすく，構成可能な決定木の大きさが限定されてしまう問題がある．Rodríguez-Cuadrado(2020) et al. [16] らは，決定木の同一階層の分枝のうち能力推定値とその分布が類似するものを統合することで，枝刈りを行う手法を提案している．同様に，Yan et al.(2004) [17] らは，回帰木を用いて項目選択する手法を提案しており，回帰木の構築時に同様なスコアの分枝を統合している．この様に，分枝を統合することで分枝数の増加を軽減し，テストの長さに関わらず決定木を構成できる．しかし，これらの手法では，能力推定値を示す分布を統合する過程で能力推定誤差が発生してしまう問題がある．

また，適応型テストにおいて，項目の露出数（出題回数）が大きい項目は，多くの受検者間で共有されやすく，経年劣化につながり，テスト全体の信頼性が低下する恐れがある [32, 33, 40]．決定木を用いた適応型テストにおいても露出数の偏りにより，一部の項目が過剰露出する問題がある．そのため，露出数の偏りを軽減することは重要な課題の一つである．この問題を軽減するために，Revuelta & Ponsoda (1998) らは，露出率の上限値を超えた項目をアイテムバンクから除外した後に情報量が最も高い項目を出する手法を提

案している [51]．また，van der Linden (1998) らは，整数計画問題を用いて各項目の露出数の上限値やテストの長さに等に制約を満たす項目集合（シャドーテストと呼ぶ）を逐次構成し，その項目集合から能力推定値に対して情報量が最も高い項目を選択する手法を提案している [10]．別のアプローチとして，van der Linden &Veldkamp らは，適格確率（受検者に出題可能な確率）を用いた項目露出制御手法を提案している．各受検者に対して，適格確率を用いてアイテムバンクから露出率の高い項目を除外することで，各項目の露出率を制限する．また，アイテムバンク内の項目パラメータに偏りがある場合，露出率を制限しながらテスト構成するときに，条件を満たす項目が不足することからテスト構成が不可能となる可能性がある．この問題に対して van der Linden & Choi (2019) らは，van der Linden & Veldkamp らの手法 [10] を拡張して，適格確率によって不適格とされた項目に対して，非常に大きな定数 M を用いてペナルティを課すことで，最適解を得るために絶対に不可欠でない限り，不適格な項目の選択を回避できる手法を提案している [42]．その他にも，Kingsbury & Zara (1989) らは，事前にアイテムバンクをランダムに複数の項目集合に分割した後に，露出数が少ない最も少ない項目集合から項目を選択する手法を提案している [49]．しかし，Kingsbury & Zara らの手法 [49] は，露出数の偏りを軽減することができるが，項目集合間の情報量における等質性は保証されないため，受検者間で能力推定誤差やテストの長さにに偏りが生じる問題がある．その問題に対し，Miyazawa & Ueno (2019) らは，情報量を制約条件として露出数が最小の項目集合を構成し，その項目集合から逐次的に情報量が最も高い項目を出題する手法を提案している [39]．この手法と同じく，Choi & Lim らは van der Linden の手法 [10] の整数計画問題に，情報量が等質となる様に制約条件を追加した手法を提案している [43, 44]．これらの手法を用いることで，特定の項目の過剰露出を防ぐことはできるが，露出と共に情報量も制限しているため，能力推定誤差が増加してしまう問題がある．この様に，露出数の減少と能力推定誤差の増加はトレードオフの関係にある．この問題を改善するために，

Miyazawa & Ueno らは，露出数と能力推定誤差のトレードオフを制御する等質適応型テストを提案している．この手法では，等質テスト構成の技術を用いてアイテムバンクを分割し，情報量が等質な項目集合を複数構成した後に，受検者ごとにランダムに項目集合を割り当て，その項目集合から情報量が最も高い項目を逐次的に出題する．それぞれが等質な項目集合から項目選択するため，受検者間の能力推定誤差を等質にしつつ，Kingsbury & Zara (1989) らの手法と同様に，アイテムバンクを分割することで，露出数の軽減にも成功した．決定木に基づく適応型テストにおいても，項目の露出制御は重要な課題である．Delgado-Gomez et al.（2019) [15] らは，線形計画問題を用いて最大露出率を制御しながら決定木を構築する手法を提案している．Delgado-Gomez et al.（2019) の手法では，受検者の能力値の事後分布と割り当てられた項目への回答パターンから，決定木の各分枝への受検者の到達確率を計算する．各分枝への受検者の到達確率をその分枝に割り当てられた項目の露出率と同等であるとみなし，線形計画問題を用いて各項目の露出率の上限値の制約を満たす項目の中で最大の情報量を持つ項目を各分枝に割り当てる．しかし，Delgado-Gomez et al.（2019) [15] らの手法においても，分枝数の増加に伴う時間 · 空間計算量の増加は実用化において重要な問題である．

以上の様な研究背景から本研究では，決定木に基づく適応型テストにおける分枝数の増加に伴う時間 · 空間計算量の問題と分枝統合に伴う能力推定誤差の問題を解決するために，決定木を用いた 2 段階適応型テストを提案する．本手法は，はじめに予測効率の高い項目選択基準を用いて事前に構成可能な最大サイズの決定木を構築する．次に，テストの前半に決定木に基づき項目選択し，この項目選択が完了した後のテスト後半にフィッシャー情報量が最大の項目を選択する．第 1 段階では，テスト序盤の能力推定誤差を考慮した項目選択基準を用いて構築された決定木から項目選択することで希薄化パラドックスの影響を小さくできると期待できる．第 2 段階では，能力推定値は真の能力値に接近しているため，フィッシャー情報量を用いて項目選択

することで高い能力推定精度が期待できる．本論では，シミュレーション実験と実データを用いた実験により決定木を用いた 2 段階適応型テストの有効性を示す．

2　項目反応理論

項目反応理論は，数理モデルを用いたテスト理論のひとつであり，近年，コンピュータ・テスティングの普及とともに多様な評価場面で活用されている [26–30]．項目反応理論の特徴としては，以下のような点が挙げられる [34–36]．

1. 測定精度の低い異質項目の影響を小さくして受検者の能力値を推定できる．
2. 異なる項目への受検者の反応を同一尺度上で評価できる．
3. 欠測データから容易にパラメータを推定できる．

項目反応理論は，正誤判定問題や多肢選択式問題など，データが正誤の 2 値となる反応データに適用されることが一般的である．このような 2 値データに適用できる項目反応モデルとしては，2 パラメータロジスティックモデル（2PLM: 2-Parameter Logistic Model）が古くから広く利用されてきた．

2.1　2 パラメータロジスティックモデル

2PLM では，能力値 $\theta \in (-\infty, \infty)$ の受検者が項目 $i \in \{1, \ldots, N\}$ に正答する確率を以下の式で表す．

$$p(u_i = 1|\theta) = \frac{1}{1 + \exp[-1.7a_i(\theta - b_i)]} \tag{1}$$

ここで，u_i は受検者が項目 i に正答するとき 1，それ以外のとき 0 の変数である．また，$a_i \in [0, \infty)$ は，項目 i の識別力パラメータと呼ばれ，どのくら

いの精度で受検者の能力値を識別できるかを示す．$b_i \in (\infty, \infty)$ は，項目 i の難易度パラメータと呼ばれ，その項目の難しさを表す．

ここで，これらのパラメータの解釈を示すために，特性の異なる複数の項目に対する 2PLM の項目反応関数（item response function:IRF）を図 1 と図 2 に示した．図では，横軸が受験者の能力値，縦軸が項目への正答確率を表す．図 1 では，識別力パラメータ a_i を三つの値に変えた場合の IRF を示した．識別力パラメータ a_i が低い項目 1 は，IRF の傾きが小さく，能力値の変化に伴う正答確率の変化が少ないことが分かる．これは項目への正誤が能力値に依存しないことを意味しており，能力測定には不適切な項目と解釈できる．一方で，識別力パラメータ a_i が高い項目 3 では，能力 $\theta = 0$ 付近で正当確率が大きく変動していることが分かる．これは，この項目が，能力 $\theta = 0$ 付近の受検者の能力を精度良く識別できることを意味する．

また，図 2 には，難易度パラメータ b_i を三つの値に代えた場合の IRF を示した．難易度パラメータ b_i が高い項目 3 は，項目 1，2 より IRF が右にシフトしていることがわかる．その結果，能力値全域において正当確率が低くなっており，正答が難しいという特性が表現されている．また，難易度パラメータ b_i は，能力値と等しいとき，すなわち，$b_j = \theta$ のとき，正当確率が 0.5 となり，その付近で項目反応関数の勾配が最も急になる．このことは，$\theta = b_i$ となる受検者の能力を精度良く評価できることを意味している．

適応型テストでは，2PLM を用いることにより，項目特性を考慮して受検者の能力値 θ を推定することができる．

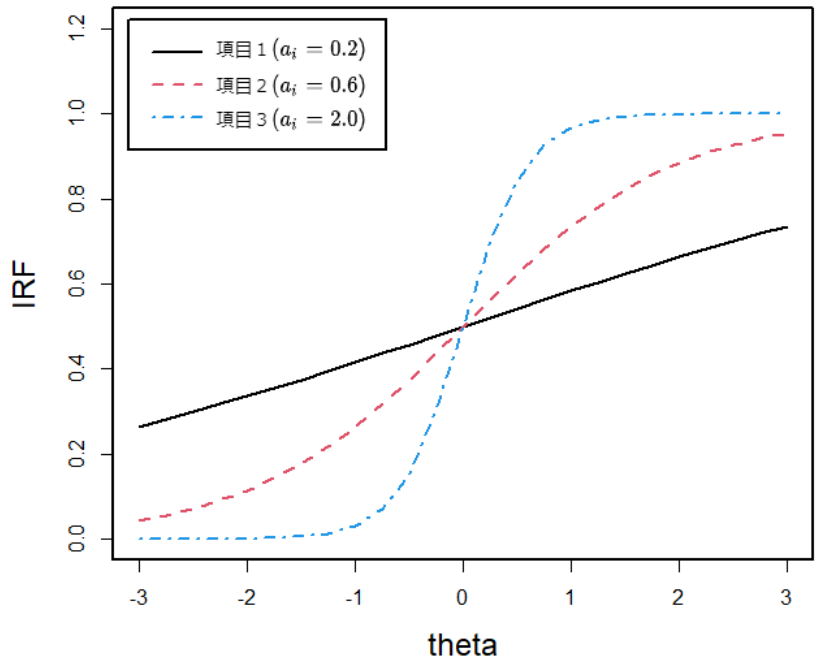

図 1　異なる識別力パラメータ a の項目に対する項目反応関数

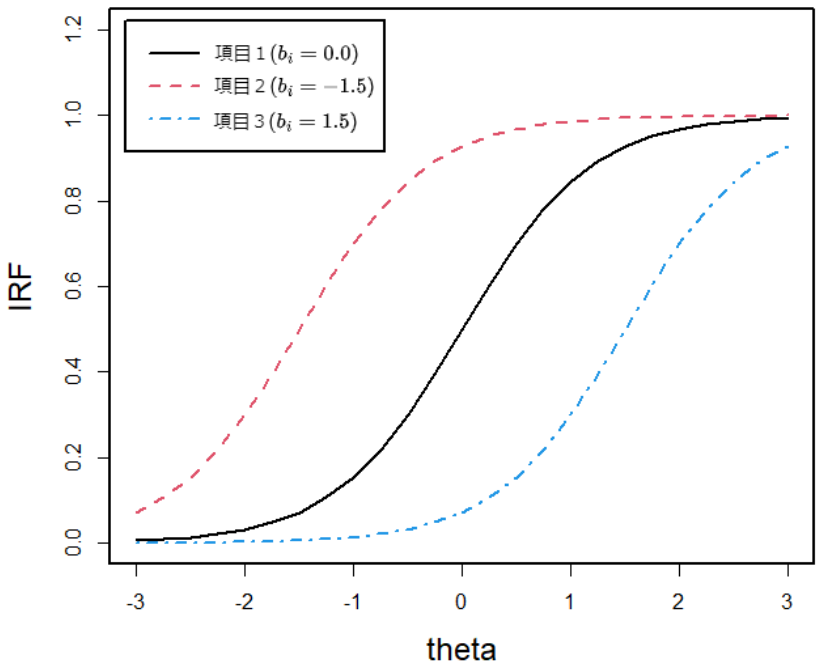

図 2　異なる識別力パラメータ b の項目に対する項目反応関数

2.2 能力値 θ の推定

受検者の能力値 θ の推定にはベイズ推定法を用いる [31]．ベイズ推定は，一致性および漸近有効性を持つと同時に少数データからの推定にも適していることが知られている [45]．本論文では，能力値 θ の推定は，$k-1$ 項目までの反応データのベクトルを u_{k-1}，能力値 θ の事前分布を $g(\theta)$ として EAP(Expected a posteriori) 推定法を用いる．EAP 推定法は，以下の事後分布の θ に関する期待値を推定値とする方法である．

$$g(\theta|u_{k-1}) = \frac{L(\theta|u_{k-1})g(\theta)}{\int L(\theta|u_{k-1})g(\theta)d\theta} \tag{2}$$

ここで，$L(\theta|u_{k-1})$ は，$k-1$ 項目の反応データを用いた能力値の尤度である．

適応型テストでは，このように受検者の能力値を推定することで，各受検者の能力値に対して情報量が最も高い項目を出題する．

2.3 フィッシャー情報量

項目反応理論においては，能力推定の標準誤差がフィッシャー情報量の逆数の値に漸近的に一致することが知られている [26]．そのため，測定精度を表す指標にフィッシャー情報量が一般的に利用される．

2PLM では，能力値 θ の受検者に対して項目 i のフィッシャー情報量を以下の式で表す [27]．

$$I_i(\theta) = \frac{[p'(u_i = 1|\theta)]^2}{p(u_i = 1|\theta)[1 - p(u_i = 1|\theta)]} \tag{3}$$

ここで，

$$p'(u_i = 1|\theta) = \frac{\partial}{\partial\theta} p(u_i = 1|\theta) \tag{4}$$

フィッシャー情報量 $I_i(\theta)$ の高い項目は，能力値 θ 付近で，その能力値をよく識別することを意味する．したがって，適応型テストでは，能力値を所

与としてフィッシャー情報量の高い項目を各受検者に出題することで，効率のよい能力測定が実現できると期待される．

なお，テストの測定精度を表す指標には，テスト T に含まれる項目集合の情報量の総和であるテスト情報量を用いる．テスト情報量は以下のように表される．

$$I_T(\theta) = \sum_{i \in T} I_i(\theta) \tag{5}$$

3　フィッシャー情報量に基づく適応型テスト

適応型テストは，受検者の能力値を逐次的に推定し，その能力値に応じて情報量が最大の項目を出題するコンピュータ・テスト手法である．各能力値に応じた項目を出題することで，能力値の測定精度を減少させずに出題項目数や受験時間を短縮できる利点があることから，本国でも近年実用化が進んでいる．本章では，適応型テストのアルゴリズムと適応型テストが抱える課題とそれに対する近年の研究を紹介する．

3.1　フィッシャー情報量に基づく適応型テストのアルゴリズム

適応型テストでは，項目特性が既知のアイテムバンクを所与として，以下の手順で受検者の能力値 θ の推定と出題する項目の選択を行う．概要は図 4 に示す．

1. 能力推定値を $\hat{\theta} = 0$ に初期化する．
2. 能力推定値 $\hat{\theta}$ を所与として情報量が最大となる項目 i をアイテムバンクから選択して受検者に出題する．
3. 項目 i に対する正誤データとそれまでの解答履歴から受検者の能力推定値 $\hat{\theta}$ を更新する．
4. 受検者の能力推定値 $\hat{\theta}$ の更新幅が閾値 ε 以下になるまで上記の手順 (2) と (3) を繰り返す．

一般的な適応型テストでは，2.3 で説明した特性から情報量としてフィッシャー情報量が用いられる．

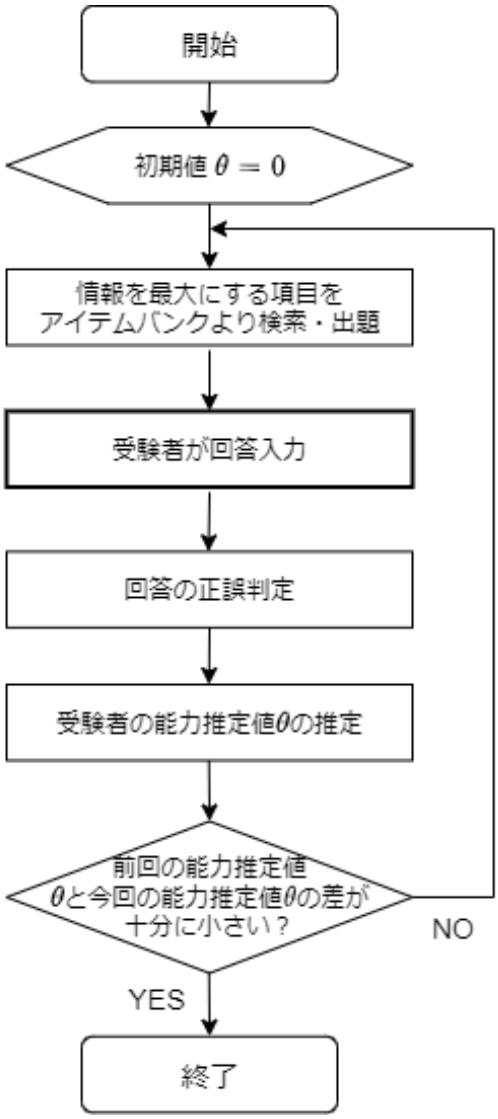

図 3　適応型テストのフローチャート

3.2　決定木に基づく適応型テスト

一般的な適応型テストでは，各受検者の能力値に対してフィッシャー情報量が最大となる項目を選択する．しかし，テスト開始直後には，能力推定値

と真の能力値が乖離しているために最適でない項目が選択される傾向がある．この問題を解決するために能力推定誤差を考慮した情報量が近年提案されている．しかし，その多くが受検者の能力パラメータ上での数値積分が必要なことから，計算コストが大きく，実用化には至っていない．その問題を解決するために決定木に基づく項目選択手法が提案されている．本節では，いくつかの決定木に基づく適応型テスト手法を紹介する．

3.2.1 Ueno & Songmuang(2010)

Ueno & Songmuang(2010) [12] は，受検者の全解答パターンに対する決定木を事前に構成し，この決定木を用いて項目選択する手法を提案している．決定木の構造は，図 4 の通りである．

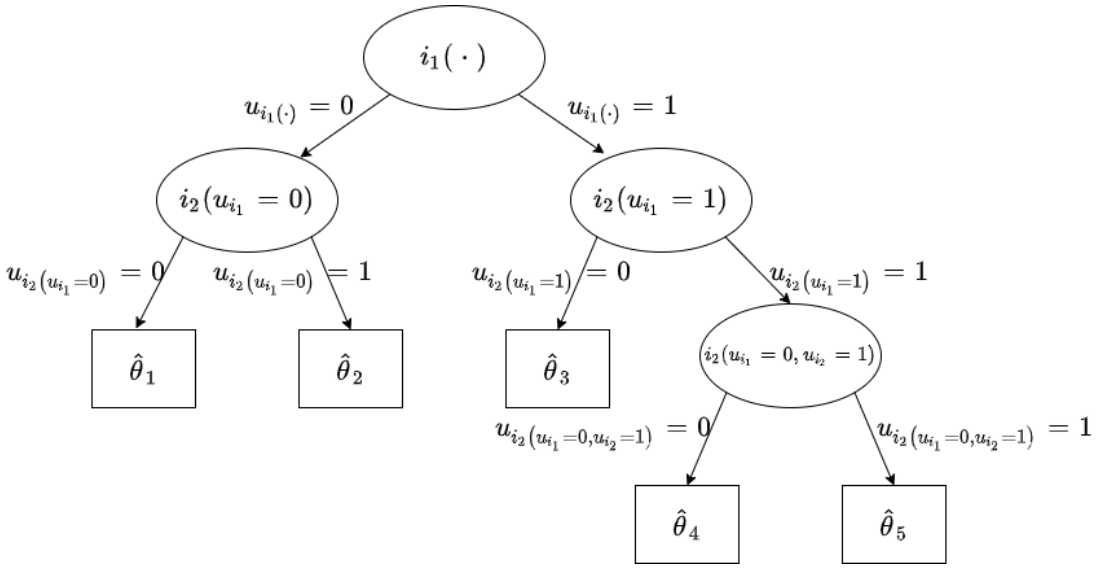

図 4　Ueno & Sonmuang(2010) で提案された決定木構造の例

$i_m\left(u_{i_1},\ldots,u_{i_{m-1}}\right)$ は，$m-1$ 問目までの解答パターンが $u_{i_1},\ldots,u_{i_{m-1}}$ である受検者に m 番目に出題される項目を表し，$u_{i_m}\left(u_{i_1},\ldots,u_{i_{m-1}}\right)$ は項目 $i_m\left(u_{i_1},\ldots,u_{i_{m-1}}\right)$ に対する解答を表し，その値は正答であれば 1，誤答であれば 0 をとる．

この手法は，事前に構築された決定木に基づいて項目を選択するために計算コストの高い項目選択基準を用いて高精度なテストを実現できる．

また，Ueno(2013) [13] らは，フィッシャー情報量に比べ，能力推定の予測効率が高い EVTI(Expected Value of Test Information) という情報量を提案している．EVTI は，EVSI(Expected Value of Sample Information) から導出されている．一般的に統計的意思決定理論では，データのもつ価値は EVSI を用いて評価される [52]．EVSI は，データ $z \in Z$ を用いた場合に予想される効用と，データがない場合に予想される効用の差で以下のように定義される．

$$\begin{aligned} EVSI = \int_Z max_{d \in D} \int_X Ut(d,x)\, p(z|x)\, p(x)\, dxdz \\ - max_{d \in D} \int_X Ut(d,x)\, p(x)\, dx \end{aligned} \tag{6}$$

ここで，$d \in D$ は行動空間 D から選ばれた行動を，$Ut(d,x)$ は x から d を選んだときの効用関数を，X は b 母数空間を示している．

EVSI のフレームワークを適応型テストの項目選択基準に適用すると，式 6 の $d \in D$ は選択される項目 $j \in R_k$ と解釈できる．適応型テストの目的は能力推定精度を上げることが目的であるため，x と $Ut(d,x)$ に θ と $\ln p(\theta|U_i1,\cdots,U_ik,U_j=0)$ を適用する．よって，テスト情報量 EVTI(Expected Value of Test Information) は以下のように定義される．

$$\begin{aligned} EVTI = max_{j \in R_k} \int_\theta [\ln p(\theta|U_i1,\cdots,U_ik,U_j=0)] \\ p(U_j=0|\theta)\, p(\theta|U_i1,\cdots,U_ik,U_j=0) + \\ [\ln p(\theta|U_i1,\cdots,U_ik,U_j=1)] \\ p(U_j=1|\theta)\, p(\theta|U_i1,\cdots,U_ik,U_j=1)\, d\theta - \\ \int_\theta [\ln p(\theta|U_i1,\cdots,U_ik)]\, p(\theta|U_i1,\cdots,U_ik)\, d\theta \end{aligned} \tag{7}$$

EVTI は能力真値 θ の受検者が項目 j を選択したときの対数予測スコアの増

加量の期待値と定義できる．しかし，EVTI の計算は受検者の能力パラメータ上で数値積分が必要なことから計算コストが非常に高い．そのため受検者の解答データに対して即時の算出が難しく，一般的な適応型テストに直接用いることは困難である．それに対し，ueno(2013) [13] らは，事前に構築された決定木を用いることによって EVTI に基づいて即時に項目選択が実現できたことを報告している．

3.2.2 Merged Tree-CAT

このように, 事前に決定木を構成することで, 即時に項目選択ができ, 高精度な適応型テストを実現できる．しかし，決定木の階層が増えるにつれ，分枝数が指数的に増加するため，計算量は $O(2^n)$ となり，時間 · 空間計算量が非常に高くなる問題がある．この問題を解決するため, Rodríguez-Cuadrado, Delgado-Gómez,and Laria(2020) [16] は，Merged Tree-CAT 法を提案している．Merged Tree-CAT 法では，同一階層の分枝の中で，推定能力値とその分布が類似している分枝を統合することで枝刈りを行い，決定木の肥大化を抑制する．具体的には, 以下の条件式 (8)(9) または (9)(10) を満たす分枝のペアを推定能力値とその分布が類似しているものとして統合する．

$$\sum_{n=1}^{Z_{m-1}} K_{i_n} > K^* \tag{8}$$

$$\left| \hat{\theta}_u^{k_s} - \hat{\theta}_v^{k_t} \right| < \frac{L_2 - L_1}{K^*} \tag{9}$$

$$\int_{-\infty}^{\infty} f_u^{k_s}(\theta) \, log \frac{f_u^{k_s}(\theta)}{f_v^{k_t}(\theta)} d\theta < \delta \tag{10}$$

条件式 (8) は統合しない状態で階層 m に存在する分枝数の合計があらかじめ設定したパラメータ K^* 以上であるか, 条件式 (9) は二つの分枝に対応する推定能力値が類似しているかどうか, 条件式 (10) は二つの分枝に対応する事

後確率分布が類似しているかどうかをそれぞれ判定している. なお条件 (8) は条件 (10) の判定にかかる計算コストを削減するための条件で, 決定木が十分に成長するまでは条件 (8) と (10) を基に分枝統合を行い, 分枝数があらかじめ定めたパラメータ値を超えた場合は決定木が十分に成長したとみなし, その後は条件 (8) と (9) を基に分枝統合を行う. パラメータ K^* を大きく設定するほど推定精度は上がり, 計算コストは増加する.

Merged Tree-CAT の分枝統合アルゴリズムを *Algorithm* 1 に示す. $nodes(m)$ は，階層 m で生成される予定の節点のリストで, それぞれが階層 $m-1$ の節点からの分岐先に対応している. 統合される分枝に対応する事後確率分布 $f_{u,v}^{k_s,k_t}$ は次のように, 受検者がそれぞれの分枝先の節点に到達する確率を用いて平均化される.

$$f_{u,v}^{k_s,k_t} = \frac{D_u^{k_s}}{D_u^{k_s} + D_v^{k_t}} + \frac{D_v^{k_t}}{D_u^{k_s} + D_v^{k_t}} \tag{11}$$

また，受検者が統合後の分枝に到達する確率 $D_{u,v}^{k_s,k_t}$ と，統合後の分枝に到達した受検者が解答した項目群 $A_{u,v}^{k_s,k_t}$ は次のように更新される.

$$D_{u,v}^{k_s,k_t} = D_u^{k_s} + D_v^{k_t} \tag{12}$$

$$A_{u,v}^{k_s,k_t} = A_u^{k_s} \cup A_v^{k_t} \tag{13}$$

これらの条件で木の成長を抑制することで, 従来手法に比べ時間 · 空間計算量の削減に成功した. 以上のように分枝を統合することで空間計算量を削減し，より大きな決定木を構成することが可能になったが，式 11 のように推定能力値が更新される際に誤差が生じる問題がある．決定木の階層が深くなり分枝数が多くなるとこの誤差は無視できない.

Algorithm 1 Merged Tree-CAT の分枝統合法

Require: m, $nodes(m)$, K^*

Ensure: $nodes(m)$

Initialisation
$length(m) :=$ 統合する前の深度 m に存在する分枝数
LOOP Process
for $i = 1, \cdots, length(m) - 1$ **do**
  for $j = i + 1, \cdots, length(m)$ **do**
    if $nodes(m)_i$の推定能力値と $nodes(m)_j$の推定能力値が類似 **then** ▷ 統合条件 (9)
      if $length(m) > K^*$ **then** ▷ 統合条件 (8)
        $nodes(m)_i$を $nodes(m)_j$に統合
        $nodes(m)_i$を $nodes(m)$ から削除
        break
      else if $nodes(m)_i$の推定能力値の分布と $nodes(m)_j$の推定能力値の分布が類似 **then**
        ▷ 統合条件 (10)
        $nodes(m)_i$を $nodes(m)_j$に統合
        $nodes(m)_i$を $nodes(m)$ から削除
        break
      end if
    end if
  end for
end for
return $nodes(m)$

3.3 各項目の露出率を考慮した適応型テスト

適応型テストにおいて，項目の露出数（出題回数）が大きい項目は，多くの受検者間で共有されやすく，経年劣化につながり，その項目の信頼性が失われやすい．上述した決定木を用いた適応型テストにおいても露出数の偏りは発生する．そのため，露出数の偏りを軽減することは重要な課題の一つである．この問題を改善するために，各項目の露出数を制限する適応型テスト手法が提案されている．

3.3.1 Restricted CAT

Revuelta & Ponsoda(1998) は，各項目の露出率の最大値を制限する手法（以下，Restricted CAT と呼ぶ）を提案した [51]．本手法は項目選択の度に各項目の露出率（=各項目の露出数/既に実施されたテストの回数）を計算し，その時点での露出率があらかじめ設定した最大露出率 r_{max} の値以下の項目群の中でフィッシャー情報量が最大となる項目を出題する．Restricted CAT は，以下の手順で項目を選択する．ここで，テストの回数を t，t 回のテスト中に項目 i が出題された回数を a_i とする．また，全てのテストにおいて十分な数の項目数を確保するために，r_{max} の値はアイテムバンクサイズとテストの長さの商の逆数より大きく設定する必要がある．

1. 能力推定値を $\hat{\theta} = 0$ に初期化する．
2. アイテムバンクから $r_{max} > a_i/t$ を満たす項目群 V を抽出．($i = 1,2,...,N$(アイテムバンクの大きさ))
3. 能力推定値 $\hat{\theta}$ を所与として情報量が最大となる項目 j を V から選択して受検者に出題する．
4. 項目 j に対する正誤データとそれまでの解答履歴から受検者の能力推定値 $\hat{\theta}$ を更新する．

5. 項目 j が出題された回数 a_j を更新する．
6. 受検者の能力推定値 $\hat{\theta}$ の更新幅が閾値 ε 以下になるまで上記の手順 (2) から (5) を繰り返す．

3.3.2 整数計画問題 (Integer Programming Problem) に基づく適応型テスト (IP)

van der Linden らは，各項目の露出数に最大露出数 R という制約を貸して項目集合（シャドーテスト）を逐次的に構成し，その中から項目選択する手法を提案している（以下，IP と呼ぶ）[11]．IP は，以下のアルゴリズムに従って項目を選択する．

1. 能力推定値を $\hat{\theta} = 0$ に初期化する．
2. 以下の整数計画問題を用いてシャドーテストを構成する．

$$
\begin{aligned}
&\text{maximise} \sum_{i=1}^{N} I_i(\hat{\theta}) x_i \qquad (14)\\
&\text{subject to}\\
&r_i x_i \leq R; (i = 1, \cdots, N),\\
&\text{(項目 } i \text{ の暴露数 } r_i, \text{ 最大暴露数 } R),\\
&\sum_{i=1}^{N} = n; \text{(テストの長さ)},\\
&x_i = \begin{cases} 1 : \text{項目 } i \text{ がシャドーテストに含まれるとき,} \\ 0 : \text{上記以外} \end{cases}
\end{aligned}
$$

3. シャドーテストから情報量が最大の項目を選択して受検者に出題する．
4. 受検者の能力推定値 $\hat{\theta}$ を更新する．
5. 受験者の能力推定値 $\hat{\theta}$ の更新幅が閾値 ε 以下となるまで上記の手順

(2) から (4) を繰り返す.

3.3.3 van der Linden and Veldkamp の適応型テスト (LV)

van der Linden & Veldkamp らは，適格確率（Eligibility probability）を用いた手法を提案している（以下，LV と呼ぶ）[4–6]．LV では，適格と判断された項目はアイテムバンクに残し，不適格と判断された項目はアイテムバンクから除外する．受検者 j に対する項目 i の適格確率を $P^j_{(E_i)}$，各項目の最大露出率をを r_{max}，受験者 j までの項目 i の露出率を a^j_i とした場合，LV は以下のアルゴリズムに従って項目を選択する．

1. 能力推定値を $\hat{\theta} = 0$ に初期化する．
2. 適格確率 $P^j_{(E_i)}$ に従ってアイテムバンクに項目を残す．不適格の場合はアイテムバンクから除外する．

$$P^j(E_i) = \min\{\frac{r_{max}}{a_i^{(j-1)}} P^{(j-1)}(E_i), 1\} \tag{15}$$

3. アイテムバンクから情報量が最大の項目を選択する．
4. 受検者の能力推定値 $\hat{\theta}$ を更新する．
5. 受験者の能力推定値 $\hat{\theta}$ の更新幅が閾値 ε 以下となるまで上記の手順 (3) から (4) を繰り返す．

ただし，$j = 1$ の場合，または $a_i^{j-1} = 0$ の場合には，$P^j_{(E_i)} = 1$ とする．また，テスト終了後には，全ての項目をアイテムバンクに戻す．

3.3.4 Big M

アイテムバンク内の項目のパラメータに偏りがある場合，項目の最大露出率を保ちながら，テストを構成することが不可能となる可能性がある．その問題に対して，van der Linden & Choi(2019) らは，前述の LV を拡張して，適格確率によって不適格とされた項目に対して非常に大きな定数 M を用い

てペナルティを課す手法（以下，Big M と呼ぶ) を提案している [42]．Big M では，3.3.2 節のテスト情報量に関する目的関数 (14) に追加項を加える．この手法は，数理計画法における標準的な手法である Big-M 法に基づいている（Williams, 1990）[54].

Big M では，下の式 16 の様な目的関数を用いる．

$$maximize \sum_{i=1}^{I} I_i(\theta)x_i - M \sum_{i \in V} x_i \tag{16}$$

ここで，M は一時的に受験者に不適格と判定された過剰露出項目の部分集合 V から項目を選択するためのペナルティの定数である．定数 M は，アイテムバンク内のすべてのアイテムに関する決定変数 x_i のすべての係数 $I_i(\theta)$ よりも十分に大きい任意の値とする（例：$M = 100$）．なお，適格性の判定には，前述の LV と同様に式 (15) が用いられる．この様な追加項を加えることで，最適解を得るために絶対に必要でない限り，過剰に露出した非適格項目を選択することを回避することができる．

3.3.5 Kingsbury and Zara (1989) の適応型テスト (KZ)

Kingsbury & Zara (1989) らは，アイテムバンクを分割することで，露出数の偏りを軽減させる手法を提案している（以下，KZ と呼ぶ）[49]．KZ では，以下のアルゴリズムに従って項目を選択する．

1. アイテムバンクをランダムに分割して複数の項目集合を構成する．
2. 能力推定値を $\hat{\theta} = 0$ に初期化する．
3. 露出数が最小の項目集合を選択し，その項目集合から情報量が最大の項目を受検者に出題する．
4. 受検者の能力推定値 $\hat{\theta}$ を更新する．
5. 受験者の能力推定値 $\hat{\theta}$ の更新幅が閾値 ε 以下となるまで上記の手順 (3) から (4) を繰り返す．

これらの手法では，露出率の偏りを軽減し，特定の項目の過剰露出を防ぐことができた．しかし，項目集合間の能力推定精度の等質性は保証されず．受検者間でのテストの長さや能力推定精度に偏りが生じる問題がある．

3.3.6 Miyazawa and Ueno の適応型テスト (MU)

この問題を解決するために，Miyazawa & Ueno らは情報量が等質になる様に制約を課した手法を提案している（以下，MU と呼ぶ）[37]．MU では，以下のアルゴリズムに従って項目を選択する．

1. 能力推定値を $\hat{\theta} = 0$ に初期化する．
2. 以下の整数計画問題を用いてシャドーテストを構成する．

$$
\begin{aligned}
&\text{minimise} \sum_{i=1}^{N} e_i x_i, (\text{項目 } i \text{ の露出数 } e_i) \qquad (17)\\
&\text{subject to}\\
&LB(\theta_l) \leq I(\theta_l) \leq UB(\theta_l); (l = 1, \cdots, L)\\
&(\text{項目 } i \text{ の暴露数 } r_i, \text{ 最大暴露数 } R),\\
&\sum_{i=1}^{N} = n; (\text{テストの長さ}),\\
&x_i = \begin{cases} 1 : \text{項目 } i \text{ がシャドーテストに含まれるとき}, \\ 0 : \text{上記以外} \end{cases}
\end{aligned}
$$

3. シャドーテストから情報量が最大の項目を選択して受検者に出題する．
4. 受検者の能力推定値 $\hat{\theta}$ を更新する．
5. 受験者の能力推定値 $\hat{\theta}$ の更新幅が閾値 ε 以下となるまで上記の手順 (2) から (4) を繰り返す．

MU では，テスト情報量の下限値と上限値を制約とすることで受験者間の

能力推定精度の等質性を保証している．しかし，これらの手法では露出数の偏りは軽減されるが，情報量を制限しているため，能力推定誤差が増加してしまう問題がある．このように，露出数の減少と，能力推定誤差の増加にはトレードオフの関係がある．

3.3.7 等質適応型テスト (UAT)

項目の露出数の減少と能力推定誤差の増加のトレードオフを制御するために，Miyazawa らは等質適応型テスト（Uniform Adaptive Testing; 以降 UAT と呼ぶ）を提案している [38, 39]．UAT では，等質テスト構成の技術を用いてアイテムバンクを分割し，情報量が等質な項目集合を複数構成し，その項目集合から項目を選択する．等質テストの構成には，石井ら (2014) [41] の手法が用いられた．石井らの手法では，等質テスト構成をグラフ上で定義される最大クリーク問題に帰着する．ここで，クリークとは，その集合に含まれる任意の頂点が全て結合された構造である．具体的には，構成されるテストを以下のグラフ構造とみなし，そのグラフの中から最大クリークの探索・抽出を行うことで，等質テストを構成する．

> 頂点：与えられたアイテムバンクにおいて，項目重複条件以外のテスト構成条件を満たすテスト（以下，テスト候補と呼ぶ）全てを頂点とする．
>
> 辺：二つのテスト候補が項目重複条件を満たしている場合，その二つの頂点（テスト候補）間に辺を引く．

このように構成されたグラフのクリークは，テスト構成条件と項目重複条件を満たす等質テスト群となる．したがって，このグラフの最大クリークを探索することで，最大の等質テスト群を構成できる．

UAT は，以下のアルゴリズムに従って項目を選択する．

0. 事前に等質テスト構成技術を用いてアイムバンクを分割して等質な項

目集合を複数構成する．

1. 受検者へ項目集合をランダムに割り当てる．
2. 能力推定値を $\hat{\theta} = 0$ に初期化する．
3. 割り当てた項目集合から情報量が最大の項目を選択して受検者に出題する．
4. 受検者の能力推定値 $\hat{\theta}$ を更新する．
5. 受験者の能力推定値 $\hat{\theta}$ の更新幅が閾値 ε 以下となるまで上記の手順 (3) から (4) を繰り返す．

このように，項目露出はテスト全体の信頼性や受検者の能力推定精度に深く関係するため，項目露出制御について様々な研究が行われている．

本研究では，露出率の上限値を基に露出率を制限する手法として代表的な Restricted CAT と Big M を実験に用いた．上述した他の項目露出制御手法の利用についても今後の課題として検討している．

4　決定木を用いた 2 段階適応型テスト

本研究では，決定木の分枝数の増加に伴う時間 · 空間計算量の問題や分枝統合の際に生じる能力推定誤差の問題を改善するため，決定木を用いた 2 段階適応型テストを提案する．本手法では，事前に予測効率の高い項目選択基準を用いて，分枝統合をせずに構成可能な最大サイズの決定木を構築する．その後，テストの 1 段階目では，事前に構築した決定木に基づき項目を選択する．つづいて，テストの 2 段階目では，1 段階目の項目選択で推定された能力値を所与としてフィッシャー情報量が最大の項目を選択する．以下で各段階についての詳細を説明する．

4.1　決定木の構築

はじめに，決定木の構築に用いる手法と，そこで採用する項目選択基準を紹介する．

4.1.1　Tree-CAT

決定木の構築には，Delgado-Gomez, Laria,and Ruiz-Hernandez(2019) [15] らが提案している項目露出を制御できる手法（以下，Tree-CAT と呼ぶ）を用いる．Tree-CAT では，受検者の能力値の事後分布と割り当てられた項目への回答パターンから，決定木の各分枝への受検者の到達確率を計算する．各分枝への受検者の到達確率をその分枝に割り当てられた項目の露出率と同等であるとみなし，線形計画問題を用いて各項目の露出率の上限値の制約を満たす項目の中で最大の情報量を持つ項目を各分枝に割り当てる．

Tree-CAT 法は，図 5 のような構造の決定木を構築する．

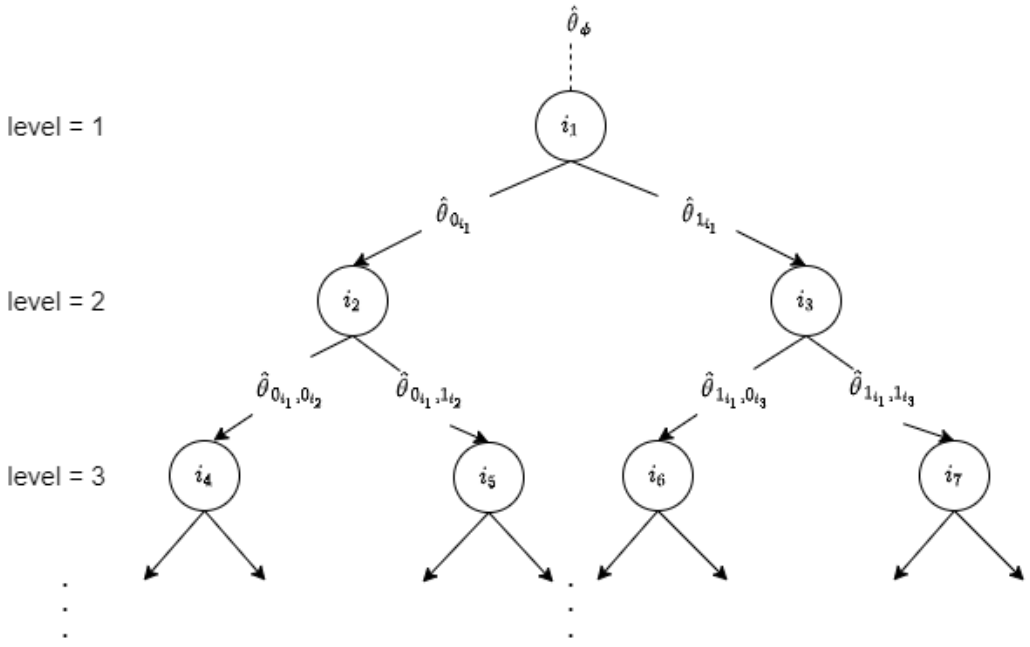

図 5　決定木の構造

各分岐先の節点に対して各項目の情報量を計算し，以下の線形計画問題を用いて各節点に割り当てる項目を選択する．

$$\max \sum_{i=1}^{N} \sum_{n=1}^{Z_m} \alpha_i^n G_i^n$$

s.t.

$$\sum_{i=1}^{N} \alpha_i^n = D_u^{k_s} \qquad n = 1, \ldots, Z_m$$

$$\sum_{n=1}^{Z_m} \alpha_i^n \leq c_i^m \qquad i = 1, \ldots, N$$

$$\alpha_i^n \geq 0 \qquad i = 1, \ldots, N \qquad n = 1, \ldots, Z_m$$

ここで，α_i^n を節点 n に項目 i が割り当てられる確率，G_i^n を節点 n に対する項目 i の情報量，$D_u^{k_s}$ を受検者が節点 n の親節点 u に到達し項目 s に対し解答 k_s を選ぶ確率，c_i^m を階層 m まで決定木を生成した後の各項目の利用可能率とする．この線形計画問題では, 事前に設定した各項目の利用可能率を超えないように制約をかけながら, 各節点での推定能力値に対して情報量が最大の項目を選択する． 具体的には以下の様な手順で決定木を構築する．

1. 受検者の能力推定値を $\hat{\theta} = 0$ に初期化する．
2. 受検者の能力推定を $\hat{\theta}$ に対応するノードを作成する．
3. 能力推定値 $\hat{\theta}$ を所与として予測効率の高い項目選択基準を用いて項目 i をアイテムバンクから選択して作成したノードに割り当てる．
4. 項目 i に対する正誤データとそれまでの解答履歴から受検者の能力推定値 $\hat{\theta}$ を更新する．
5. 決定木の階層が設定された値 L となるまで上記の手順 (2) から (4) を繰り返す．

Algorithm 2 決定木構築アルゴリズム

procedure MAKE TREE(I, L, r_{max})
  $f_0(\hat{\theta}) \coloneqq N(0,1)$
  $c \coloneqq 1$, $P \coloneqq 0_{(I\times 1)}$, $D \coloneqq r_{max\,(I\times 1)}$, $F \coloneqq \emptyset$, $k \coloneqq 0$
  $E \coloneqq CalculateInformation(f(\hat{\theta}_0))$
  while $c > 0$ **do**
    $i \coloneqq argmax\{E\}$
    $P_i \coloneqq min\{c, D_i\}$
    $c \coloneqq c - P_i$
    $D_i \coloneqq D_i - P_i$
    $F \coloneqq F \cup i$
    $level_1.node_{k+1}.item \coloneqq i$
    $level_1.node_{k+1}.Distribution \coloneqq f(\hat{\theta}_0)$
    $level_1.node_{k+1}.P \coloneqq P_i$
    $E_i \coloneqq 0$
  end while
  for $l \leftarrow 2$ to L **do**
    $K \coloneqq |F|$
    $P \coloneqq (0)_{I\times K}$
    $F \coloneqq \{F_1, \cdots, F_K\}$, $F_k \coloneqq \emptyset \; \forall \, k = 1, \cdots, K$
    while $k \leq K$ **do**
      $c \coloneqq C_k$
      while $c > 0$ **do**
        while $r \leq 2$ **do**
          $E_k^r \coloneqq CalculateInformation(f_k(\hat{\theta}))$
          while $i \leq I$ **do**
            if $D_i == 0$ **then**
              $E_i \coloneqq 0$
            end if
          end while
          $i_k^r \coloneqq argmax\{E_{ik}^r\}$
          $P_{ik}^r \coloneqq min\{C_k, D_i\}$
          $D_i \coloneqq D_i - P_{ik}^r$
          $c \coloneqq c - P_{ik}^r$
          $F_k \coloneqq F_k \cup i_k^r$
          $f_{u^k}^{i_k^r}(\hat{\theta}) \coloneqq UpdateDistribution(f_k(\hat{\theta}),\, u_k,\, i_k^r,\, r)$
          $level_l.node_k^r.item \coloneqq i_k^r$
          $level_l.node_k^r.Distribution \coloneqq f_{u^k}^{i_k^r}(\hat{\theta})$
          $level_l.node_k^r.P \coloneqq P_{ik}^r$
          $E_{ik}^r \coloneqq 0$
        end while
      end while
    end while
  end for
  $T \coloneqq \{level_1, \cdots, level_{L+1}\}$
  return T
end procedure

具体的なアルゴリズムを *Algorithm* 2 に示す．本アルゴリズムでは，はじめに根節点 $level_1.node$ を生成する (2 ˜15 行目)．はじめに，受検者の推定能力値の事前分布を $f_0(\hat{\theta})$ と項目が割り当てられていない受験者の割合 c，項目 i の利用率 P_i の配列 P，項目 i の露出可能率 D_i の配列 D を初期化する (2˜3 行目)．4 行目で，推定能力値の事前分布に対してアイテムバンク I 内の全ての項目に対して項目 i が持つ情報量 E_i を計算する (4 行目)．項目が割り当てられていない受験者の割合 $c = 0$ となるまで K 個の根節点が生成される (5˜15 行目)．各節点は，割り当てられた項目 $level_1.node.item$ とその節点に到達した受験者の推定能力値の分布 $level_1.node.Distribution$，その節点への受検者の到達確率 $level_1.node.P$ を保持している (11˜13 行目)．一度選択された項目が再び選択されないように $E_i := 0$ とする (14 行目)．K 個の根節点が生成されると，L 階層まで各根に対応する木が反復的に生成される (16˜43)．項目 i の露出可能率 $D_i = 0$ となった場合，項目 i の情報量 $E_i := 0$ として，再び選択されることを防ぐ (25 ˜29 行目)．節点 k において回答が r であった場合に対応する分枝に対して選択される項目を i_k^r，節点 k に到達した受検者のこれまでの回答パターンを u_k に基づき，推定能力値の事後分布 $f_{u^k}^{i_k^r}(\hat{\theta})$ を更新する (35 行目)．その他は根節点の生成と同様である．

4.1.2 能力推定誤差を考慮した項目選択基準

適応型テストでは，一般的に項目選択基準として MFI(Maximum Fisher Information) [26, 46] を採用しており，能力推定値に対してフィッシャー情報量が最大の項目を選択する．しかし，テスト開始直後には，能力推定値と真の能力値が乖離しているために最適でない項目が選択される傾向がある．これは希薄化パラドックス（attenuation paradox）として古くから知られている [27]．この問題を解決するためにいくつかの項目選択基準が提案されている．ここでは，本論文で扱ういくつかの能力推定誤差を考慮できる項目選択基準を紹介する．

4.1.3 Maximum Posterior-Weighted Information(MPWI)

van der Linden and Pashley(1998) は，能力推定値の事後分布が十分に収束していない状態で，ある一点の推定値においてフィッシャー情報量が最大となる項目を選択することは，その近傍の能力パラメータの尤度を無視することになり，最適な項目選択とは言えないとして，フィッシャー情報量を能力値の事後分布で重みづけした情報量 Maximum Posterior-Weighted Information(MPWI) を提案した．

$$MPWI = \max_{j \in R_k} \int J_{U_j}(\theta) g(\theta | u_{i_1}, ..., u_{i_{k-1}}) d\theta \tag{18}$$

ここで，

$$J_{u_{i_1},...,u_{i_{k-1}}}(\theta) = -\frac{\partial}{\partial \theta^2} \ln L(\theta | u_{i_1}, ..., u_{i_{k-1}}) \tag{19}$$

$$L(\theta | u_{i_1}, ..., u_{i_{k-1}}) = \prod_{j=1}^{k-1} \frac{exp[a_{i_j}(\theta - b_{i_j})]^{u_{i_j}}}{1 + exp[a_{i_j}(\theta - b_{i_j})]} \tag{20}$$

$$g(\theta | u_{i_1}, ..., u_{i_{k-1}}) = \frac{L(\theta | u_{i_1}, ..., u_{i_{k-1}})}{\int L(\theta | u_{i_1}, ..., u_{i_{k-1}}) d\theta} \tag{21}$$

ここで，u_{i_k} は k 番目に出題した項目とそれに対する回答パターンとする．(19) は観測情報量と呼ばれるが，項目パラメータが既知の場合，フィッシャー情報量と観測情報量は等しいとされている．等式の導出は（Veerkamp, 1996）[9] で示されている．

4.1.4 Minimum Expected Posterior Variance(MEPV)

van der Linden and Pashley(2009) は，能力値についての事後分布の分散が最小になる項目を選択する Minimum Expected Posterior Variance(MEPV) を

提案した．回答パターンが正誤の 2 パターンのテストの場合，MEPV は以下のように定義される．

$$\begin{aligned} MEPV = \min_{i \in R_n} p_i(r=0|u_{n-1}) \int p(\theta|u_n)(\theta - \hat{\theta_{u_n}})^2 d\theta \\ + p_i(r=1|u_{n-1}) \int p(\theta|u_n)(\theta - \hat{\theta_{u_n}})^2 d\theta \end{aligned} \tag{22}$$

ここで，r は受験者の回答パターン（正答の場合は $r=1$，誤答の場合は $r=0$），u_n はこれまでに出題された n 個の項目とそれに対する回答パターン，R_n は n 番目の項目を選択するときに選択可能な項目群とする．

これらの項目選択基準は，前述したフィッシャー情報量の問題を解消し，受検者の能力推定の信頼性を向上できた．しかし，これらの項目選択基準は，能力パラメータ上で数値積分が必要であるため，極めて高い計算コストが要求される．受検者の回答の度に情報量を計算する一般的な適応型テストでは即時に項目選択することが難しいため，これらの項目選択基準は実用的ではない．しかし，事前に受検者の回答パターンに対する項目決定木を構築することで，これらの項目選択基準も実用可能になる．

4.2 1 段階目における決定木を用いた項目選択

2 段階適応型テストの 1 段階目では，以下の様な手順で事前に構成した決定木に基づき項目選択を行う．

1. 受検者に決定木の第一階層の節点に対応する項目を出題する．
2. 受検者の解答に応じて分枝の先にある節点を受検者に割り当てる．
3. 受験者に割り当てられた節点に対応する項目を受検者に出題する．
4. 決定木の末の節点まで上記の手順 (2) と (3) を繰り返す．

具体的なアルゴリズムを *Algorithm* 3 に示す．事前に構成した決定木 T には，決定木の各階層の節点の情報が格納されている．R はアイテムバンク内

Algorithm 3 1 段階目における決定木を用いた項目選択アルゴリズム

1: **procedure** ITEM SELECTION IN TREE(T, R, L)
2: $\hat{\theta}_0 := 0$
3: **if** $numberofT.level_l.node_\emptyset == 1$ **then**
4: $next.item := T.level_1.node_\emptyset.item$
5: **else**
6: $T.level_1.node_\emptyset.P$ の確率で分枝 k を選択
7: $next.item := T.level_1.node_{\emptyset_k}.item$
8: **end if**
9: $u_1 := R_{next.item}$
10: **if** $number\ of\ T.level_2.node_{u_1} == 1$ **then**
11: $next.node := T.level_2.node_{u_1}$
12: **else**
13: $T.level_2.node_{u_{1k}}.P$ の確率で分枝 k を選択
14: $next.node := T.level_2.node_{u_{1k}}$
15: **end if**
16: $\hat{\theta}_1 :=$ GetExpectedValue($next.node.Distribution$)
17: **for** $l \leftarrow 2$ to L **do**
18: $next.item := next.node.item$
19: $u_l := u_{l-1} \cup R_{next.item}$
20: **if** $numberofT.level_{l+1}.node_{u_l} == 1$ **then**
21: $next.node := T.level_{l+1}.node_{u_l}^{R_{next.item}}$
22: **else**
23: $T.level_{l+1}.node_{u_{lk}}.P$ の確率で分枝 k を選択
24: $next.node := T.level_{l+1}.node_{u_{lk}}^{R_{next.item}}$
25: **end if**
26: $\hat{\theta}_l :=$ GetExpectedValue($next.node.Distribution$)
27: **end for**
28: return $\{\hat{\theta}_L, u_L\}$
29: **end procedure**

の全項目に対する受検者の回答パターンデータ，L は決定木の階層数とする．はじめに，受検者の推定能力値を $\hat{\theta}_0$ を初期化する (2 行目)．続いて，根節点が一つの場合は，その節点に割り当てられた項目を出題する (3,4 行目)．露出率を制限する場合には，根節点が複数存在するため，各節点への到達確率 $T.level_1.node_\emptyset.P$ に従い節点を選択する (5~8 行目)．出題された項目への

受検者の回答パターン u_1 に対応する分枝が一つの場合は，その分枝に対応する節点を次の節点とする (10,11 行目)．回答パターン u_1 に対応する分枝が複数ある場合は，各分枝先の節点への到達確率 $T.level_2.node_{u_{1k}}.P$ に従い次の節点を選択する (12 ~15 行目)．次の節点が決定され次第，その節点に対応する推定能力値の事後分布から推定能力値を求め，$\hat{\theta}_1$ を更新する (16 行目)．受検者が決定木の葉節点へ到達するまで，同様の工程を繰り返す (17 ~27 行目)．

4.3　2 段階目におけるフィッシャー情報量に用いた項目選択

決定木による項目選択が完了した後は，図 6 のように 2 段階目として決定木の葉節点に対応する能力推定値を所与として，アイテムバンクからフィッシャー情報量が最大の項目を選択する一般的な CAT へ切り替える．切り替え後は以下の様な手順で項目が選択される．

1. 能力推定値を $\hat{\theta} = \hat{\theta_L}$ に初期化する ($\hat{\theta_L}$ は，決定木の葉節点に対応する推定能力値)．
2. フィッシャー情報量が最大となる項目を選択する．
3. 選択された項目への反応データとそれまでの解答履歴から能力推定値 $\hat{\theta}$ を更新する．
4. 手順 (2) と (3) を，テストの終了条件まで繰り返す．

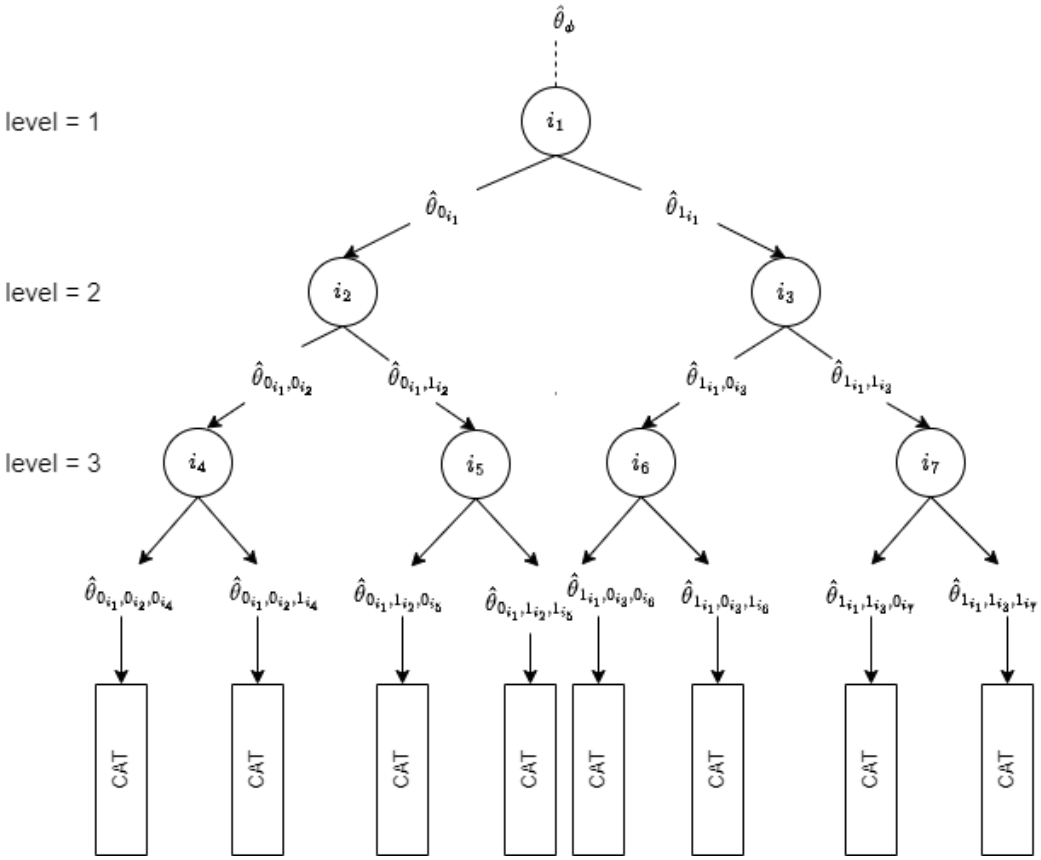

図 6　決定木から CAT への切り替え

Algorithm 4 2 段階目におけるフィッシャー情報量を用いた項目選択アルゴリズム

procedure ITEM SELECTION IN CAT(I, $\hat{\theta}_L$, U, R, L_{CAT})
    $\hat{\theta}_0 := \hat{\theta}_L$
    for $l \leftarrow 1$ to L_{CAT} **do**
        $E :=$ CalculateFisherInfomation(I, $\hat{\theta}_{l-1}$)
        $E_U := 0$
        $next.item := argmax\{E\}$
        $U := U \cup next.item$
        $u := \{U, R_U\}$
        $\hat{\theta}_l :=$ GetExpectedValue($f_u(\hat{\theta})$)
    end for
    return $\hat{\theta}_{L_{CAT}}$
end procedure

具体的なアルゴリズムを *Algorithm* 4 に示す．はじめに，受検者の推定能力値を $\hat{\theta}_0$ を初期化する (2 行目)．続いて，推定能力値に対してアイテムバンク内の各項目のフィッシャー情報量 E を計算する (4 行目)．回答済みの項目群 U に含まれる項目の情報量は $E_U := 0$ とし，再び選択されることを防ぐ (5 行目)．回答済みの項目群 U と，それらの項目に対する受検者の回答 R_U からなる回答パターンデータを u とする (8 行目)．回答パターン u を基に推定能力値を更新する (9 行目)．以上の工程をテスト終了条件まで繰り返す (3 ~10 行目)．なお，今回はテスト終了条件を出題項目数 L_{CAT} とした．

このように，テスト前半に能力推定誤差を考慮した項目選択基準に基づいて構成された決定木を用いて項目選択し，ある程度項目が出題されて能力推定誤差が小さくなったテスト後半に，計算コストが低く，能力推定値の漸近的誤差に収束するフィッシャー情報量に基づく項目選択基準に切り替えることで，決定木の分枝数の増加に伴う時間 · 空間計算量の問題と，テスト全体の能力推定精度の改善を図る．

5　評価実験

本章では提案手法の有効性を示すため，評価実験を行う．最初に，決定木構築時に採用する項目選択基準と決定木からの CAT への切り替えタイミングを決定するために，MFI，MPWI，MEPV，EVTI を比較する．次に，CAT と Tree-CAT，Merged Tree-CAT と Tree-CAT から CAT へ切り替える提案手法の能力推定精度 (RMSE) と露出率の平均値と最大値を比較する．最後に，各項目の最大露出率を制約として，Restricted CAT と Big M，Tree-CAT，Merged Tree-CAT と，Tree-CAT から Restricted CAT に切り替える提案提案手法と Big M 法に切り替える提案手法の能力推定精度 (RMSE) と露出率の平均値と最大値を比較する．以上の実験から提案手法の有効性を示す．なお，本論文の実行環境は Ubuntu18.04 を OS とする計算機 (CPU: Intel Core i9-9900X 3.50 GHz, RAM 128 GB) である．

5.1 反応パタンの生成

本論文の実験には，100，500，1000，2000 項目のシミュレーションデータからなるアイテムバンクを用いた．シミュレーション実験の手順は以下の通りである．

1. 以下の特徴を持つアイテムバンクを生成する．
   （a）項目数 $N = 100, 500, 1000, 2000$
   （b）識別力パラメータを $a \in U(0,1)$ からサンプリングする．
   （c）難易度パラメータを $b \in U(-4,4)$ からサンプリングする
2. 受検者の能力真値を $\theta \sim U(-4,4)$ からサンプリングする．
3. 受検者の能力推定値を $\hat{\theta} = 0$ に初期化する．
4. 各手法に応じて項目を選択する．
5. 項目への反応データを，能力真値と項目パラメータを所与として発生させる．
6. 項目への反応データとそれまでの解答履歴から能力推定値 $\hat{\theta}$ を求める．
7. テスト終了条件を満たすまで手順 (4) から (7) を繰り返す．
8. 手順 (2) から (8) を 1000 回繰り返す．生成された出題パターンと解答履歴を用いて，a）能力推定値の RMSE，b）各項目の露出率の平均値と最大値に関する統計量を求めた．

シミュレーションアイテムバンクの設定や受検者の能力真値の設定は Ueno & Ponsoda (2010) [12]， Linden (1998) [10] に基づく．テスト終了条件については，能力推定値が能力真値に収束する前に能力推定値の更新幅が小さくなってしまう可能性があるため，本実験では出題項目数とした．

さらに，978 項目の実データからなる実アイテムバンクも用いて実験を行い提案手法の有効性を評価する．ここでは，リクルート（株）で開発された

SPI [53] のアイテムバンクを用いて，上記と同様の手順で実験した．実アイテムバンクの詳細は表 1 の通りである．

表 1　実アイテムバンクの詳細

Pool Size	Parameter a				Parameter b			
	Min	Max	Mean	SD	Min	Max	Mean	SD
978	0.12	3.08	0.43	0.2	-4	4.55	-0.22	1.16

5.2　パラメータチューニング

本節では，決定木を構成する際に採用する項目選択基準と，決定木を用いた項目選択からフィッシャー情報量を用いた項目選択への切り替えタイミングについて検討する．ここでは，決定木からフィッシャー情報量を用いた項目選択へ切り替えるタイミングを 1 項目目から 14 項目目まで変更し，前述の手順でシミュレーション実験を実施した．本実験では，構成可能できた決定木の最大値が 13 項目までであったため，切り替えタイミングの上限値を 14 としている．決定木における項目選択基準は MFI，MEPV，MPWI，EVTI とした．なお，テストの出題項目数は 30 項目とし，サイズ 1000 のシミュレーションアイテムバンクを使用した．結果を図 7 に示す．図 7 は横軸が切り替え時の出題項目数であり，縦軸に能力真値と能力推定値の RMSE を示した．図 7 から MEPV が最も RMSE が小さくなった．また，MEPV は切り替えタイミングが遅い程 RMSE が小さくなる傾向があった．よって本実験では決定木における項目選択基準は MPWI を採用し，切り替えタイミングは 14 項目（13 項目まで決定木を用いて項目選択，14 項目からと一般的な適応型テストを用いて項目選択）とする．

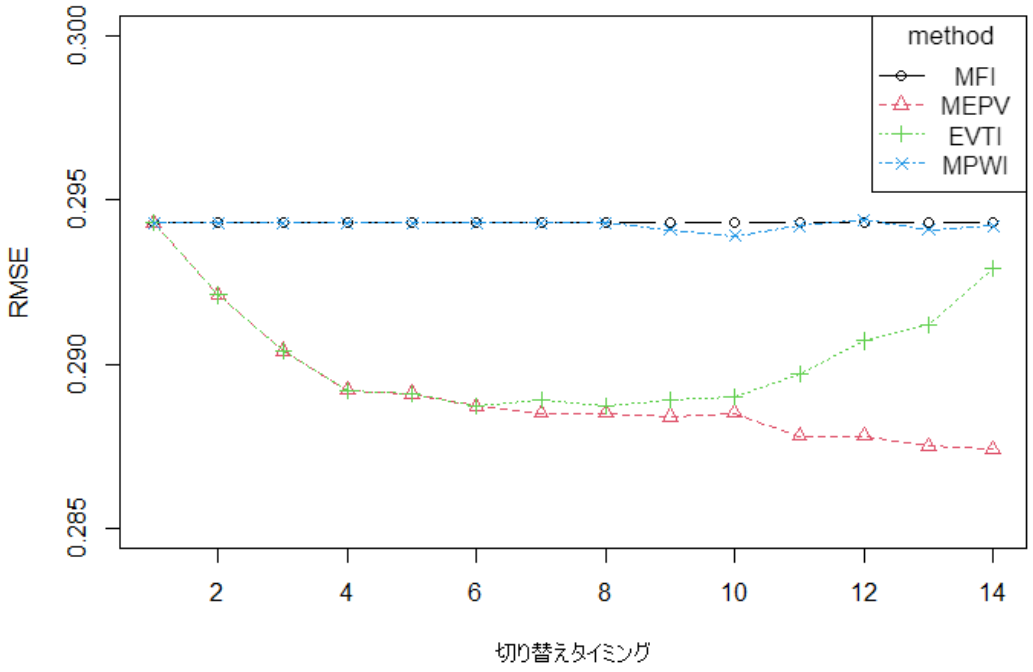

図 7　切り替えタイミングに対する各項目選択基準の RMSE

5.3　実験結果

表 2 は，アイテムバンクサイズが 100,500,1000,2000 のシミュレーションアイテムバンクを使用したときの CAT と，Tree-CAT，Merged Tree-CAT，提案手法の推定能力値の RMSE を示している．また，表 3 は各条件での露出率の平均値と最大値を示している．なお，平均値については 1 回以上露出した項目から算出した．

Tree-CAT は，テストの長さが 14 以上の場合はメモリオーバーにより，決定木を構成できなかった．これは，分枝数の指数的増加に伴う時間 · 空間計算量の問題が原因である．アイテムバンクサイズや出題項目数に関わらず，提案手法の方が Merged Tree-CAT と比べて能力推定精度が高かった．また，出題項目数が多くなるほど Merged Tree-CAT と提案手法の能力推定精度の

表 2　シミュレーションデータを用いたときの各手法の RMSE

アイテム バンク	出題 項目数	CAT (MFI)	Tree (MEPV)	M Tree (MEPV)	提案 (MEPV,MFI)
100	13	0.587	0.566	0.570	0.566
	30	0.426	—	0.452	0.419
	50	0.393	—	0.420	0.391
	80	0.388	—	—	0.388
500	13	0.516	0.498	0.503	0.498
	30	0.311	—	0.326	0.298
	50	0.242	—	0.275	0.236
	80	0.203	—	0.250	0.199
1000	13	0.503	0.490	0.494	0.490
	30	0.294	—	0.305	0.287
	50	0.222	—	0.248	0.218
	80	0.180	—	0.216	0.178
2000	13	0.497	0.486	0.491	0.486
	30	0.277	—	0.297	0.273
	50	0.206	—	0.202	0.203
	80	0.163	—	0.195	0.163

差が顕著に見られた．これは，Merged Tree-CAT は節 3.2.2 でも述べたように，分枝を統合する過程で能力推定値に誤差が生じる問題が要因の一つであると考えられる．さらに，Merged Tree-CAT は分枝を統合する際に，それぞれの分枝に達した受験者が既に回答した項目群を 3.2.2 節の式 (13) のように統合する．これにより，それぞれの分枝に達した受験者が回答した項目群が完全に同一である場合を除いて，既に回答した項目群すなわち以後出題不可能な項目群が大きくなる．表 3 を見ると，Merged Tree-CAT の露出率の平

表3　シミュレーションデータを用いたときの各手法の露出率の平均値（最大値）

アイテム バンク	出題 項目数	CAT (MFI)	Tree (MEPV)	M Tree (MEPV)	提案 (MEPV,MFI)
100	13	0.295	0.266	0.265	0.266
		(1.0)	(1.0)	(1.0)	(1.0)
	30	0.385	—	0.370	0.381
		(1.0)		(1.0)	(1.0)
	50	0.550	—	0.515	0.544
		(1.0)		(1.0)	(1.0)
	80	0.862	—	—	0.861
		(1.0)			(1.0)
500	13	0.176	0.143	0.142	0.143
		(1.0)	(1.0)	(1.0)	(1.0)
	30	0.184	—	0.132	0.156
		(1.0)		(1.0)	(1.0)
	50	0.215	—	0.173	0.202
		(1.0)		(1.0)	(1.0)
	80	0.251	—	0.214	0.243
		(1.0)		(1.0)	(1.0)
1000	13	0.148	0.124	0.124	0.124
		(1.0)	(1.0)	(1.0)	(1.0)
	30	0.149	—	0.103	0.138
		(1.0)		(1.0)	(1.0)
	50	0.177	—	0.106	0.170
		(1.0)		(1.0)	(1.0)
	80	0.203	—	0.131	0.197
		(1.0)		(1.0)	(1.0)
2000	13	0.134	0.125	0.124	0.125
		(1.0)		(1.0)	(1.0)
	30	0.123	—	0.109	0.118
		(1.0)		(1.0)	(1.0)
	50	0.134	—	0.102	0.131
		(1.0)		(1.0)	(1.0)
	80	0.152	—	0.095	0.152
		(1.0)		(1.0)	(1.0)

均値がその他の手法に比べて低くなっている．これは，多くの分枝で出題されやすい良質な項目が，出題不可能な項目群に含まれやすくなったことが原因であると考えられる．このような出題不可能な項目群の増加も，Merged Tree-CAT の能力推定精度の低下の要因の一つであると思われる．出題項目数が多くなるほど分枝の統合回数が増えるため，これらの分枝統合による問題の影響は大きくなる．提案手法は全ての項目を決定木から出題する必要がないため，分枝統合による能力推定精度悪化の影響を受けずに高い能力推定精度を維持できている．

また，CAT と比べても提案手法は能力推定精度が高かった．これは，テスト序盤に項目選択基準として能力推定誤差を考慮可能な MEPV を採用し，能力推定誤差が小さくなったテスト後半に MFI に切り替えることで，テスト序盤の予測精度を向上させながら，情報量の高い項目をテスト後半に温存できたことが要因であると考えられる．しかし，Merged Tree-CAT 程の能力推定精度の差は見られなかった．これは，露出率を制限していないため，項目露出の偏りが大きい MFI を用いても，制限されることなく情報量の高い項目を出題できたたためであると思われる．

露出率については，最大値はどの条件でも 1.0 と変わらなかったが，平均値は CAT が高い傾向が見られた．これは，CAT は項目選択基準として MFI のみを使用しているため，項目選択の偏りが大きいことが要因であると思われる．一方で，提案手法は MFI と項目選択の偏りが小さい MEPV を併用しているため，CAT に比べ露出率の平均値を比較的低く抑えることができた．

アイテムバンクサイズについては，大きいほど情報量の高い項目が多く存在するため能力推定精度が高くなっている．また，100 項目のアイテムバンクでは提案手法に比べ Merged Tree-CAT 法の能力推定精度が特に悪く，さらに 80 項目まで出題することができなかった．前述の通り，Merged Tree-CAT 法は分枝統合を繰り返すことで，出題可能な項目数が減少していくため，100 項目程度の小さなアイテムバンクの場合，その問題が顕著に現れていることが能力推定精度の悪化の原因であると思われる．

続いて，実データからなるアイテムバンクを用いて同様の実験を行った．表 4 は推定能力値の RMSE を表し，表 5 は露出率の平均値と最大値を表す．今回使用した実データは，シミュレーションデータよりも，識別力パラメータの値が比較的低いため，能力推定値もシミュレーションの結果に比べて低くなっている．実データにおいても提案手法は CAT や Merged Tree-CAT 法よりも能力推定精度を改善できていた．露出率を見ると，最大値はシミュレーションデータと同様にどの手法においても 1.0 であった．平均値はシミュレーションデータを用いた場合よりも比較的高くなっていた．これは，シミュレーションデータよりも識別力パラメータが極端に高い項目が含まれることから，項目選択に偏りが生じたことが要因であると考えられる．

表 4　実データを用いたときの各手法の RMSE

アイテム バンク	出題 項目数	CAT (MFI)	Tree (MEPV)	M Tree (MEPV)	提案 (MEPV,MFI)
978	13	0.539	0.532	0.534	0.532
	30	0.356	——	0.360	0.351
	50	0.281	——	0.302	0.279
	80	0.240	——	0.276	0.236

表 5　実データを用いたときの各手法の露出率の平均値（最大値）

アイテム バンク	出題 項目数	CAT (MFI)	Tree (MEPV)	M Tree (MEPV)	提案 (MEPV,MFI)
978	13	0.213	0.194	0.194	0.194
		(1.0)	(1.0)	(1.0)	(1.0)
	30	0.191	—	0.140	0.152
		(1.0)		(1.0)	(1.0)
	50	0.198	—	0.139	0.178
		(1.0)		(1.0)	(1.0)
	80	0.217	—	0.157	0.208
		(1.0)		(1.0)	(1.0)

5.4　露出率を制約とした項目選択における実験結果

続いて各項目の露出率を制限した場合について実験を行う．3.3.6 節で述べたように，露出率の減少と能力推定誤差の増加はトレードオフの関係があるため，能力推定精度を比較するためには，露出率を揃えることが望ましい．しかし，実際の露出率は全ての受検者に対してテストを実施するまで分からないため，異なる露出制御手法間で露出率を完全に揃えることは困難である．そのため，本実験では各手法において項目露出率の最大値を $r_{max} = 0.3$ と制約して実験を行った．本実験では Restricted CAT(RCAT), Big M，Tree-CAT(Tree)，Merged Tree-CAT(M Tree)，決定木からの切り替え後に Restricted CAT を採用した提案手法 (提案 RCAT)，決定木からの切り替え後に Big M を採用した提案手法 (提案 BigM) の 6 つの手法を比較した．その他の条件は上述の実験と同様とする．表 6 は各手法の能力推定値の RMSE，表 7 は各手法の平均露出率と最大露出率を示している．

Tree-CAT は前述の露出率を制限しない場合と同様に，テストの長さが 14 以上の場合はメモリオーバーにより，決定木を構成することができなかった．また，アイテムバンクサイズ $N = 100$ の場合，Big M と決定木から Big

M へ切り替える提案手法以外は 30 項目以上のテストを構成することができなかった．Restricted CAT と Merged Tree-CAT，決定木から Restricted CAT へ切り替える提案手法は，予め設定した最大露出率 r_{max} の制約を満たす項目が不足したためである．一方，Big M と決定木から Big M へ切り替える提案手法は，30 項目以降もテストを構成することができた．これは，予め設定した最大露出率を超えた項目が完全に選択不可能となる Restricted CAT と違い，Big M は最大露出率を超えてペナルティが課された項目も選択することができるためである．

表 6　シミュレーションデータを用いて露出率を制約とした各手法における RMSE

アイテムバンク	出題項目数	RCAT (MFI)	BigM (MFI)	Tree (MEPV)	M Tree (MEPV)	提案 RCAT (MEPV,MFI)	提案 BigM (MEPV,MFI)
100	13	0.847	0.596	0.589	0.595	0.589	0.589
	30	—	0.520	—	—	—	0.464
	50	—	0.389	—	—	—	0.377
	80	—	0.352	—	—	—	0.345
500	13	0.558	0.520	0.513	0.517	0.513	0.513
	30	0.365	0.342	—	0.336	0.347	0.334
	50	0.340	0.305	—	0.294	0.329	0.293
	80	0.322	0.279	—	0.275	0.308	0.272
1000	13	0.535	0.512	0.505	0.508	0.505	0.505
	30	0.329	0.317	—	0.312	0.316	0.302
	50	0.283	0.272	—	0.272	0.267	0.260
	80	0.247	0.241	—	0.241	0.229	0.226
2000	13	0.532	0.501	0.491	0.493	0.491	0.491
	30	0.305	0.294	—	0.293	0.282	0.276
	50	0.239	0.230	—	0.244	0.221	0.220
	80	0.194	0.193	—	0.232	0.182	0.182

表 7　シミュレーションデータを用いて露出率を制約とした各手法における露出率の平均値（最大値）

アイテムバンク	出題項目数	RCAT (MFI)	BigM (MFI)	Tree (MEPV)	M Tree (MEPV)	提案 RCAT (MEPV,MFI)	提案 BigM (MEPV,MFI)
100	13	0.185	0.227	0.219	0.217	0.219	0.219
		(0.299)	(0.358)	(0.341)	(0.344)	(0.341)	(0.341)
	30	――	0.300	――	――	――	0.304
			(0.395)				(0.404)
	50	――	0.500	――	――	――	0.500
			(0.999)				(0.998)
	80	――	0.800	――	――	――	0.800
			(1.0)				(1.0)
500	13	0.114	0.130	0.124	0.123	0.124	0.124
		(0.299)	(0.336)	(0.339)	(0.340)	(0.339)	(0.339)
	30	0.125	0.138	――	0.133	0.129	0.136
		(0.299)	(0.345)		(0.349)	(0.331)	(0.352)
	50	0.141	0.152	――	0.156	0.136	0.144
		(0.299)	(0.349)		(0.353)	(0.329)	(0.355)
	80	0.178	0.192	――	0.195	0.170	0.187
		(0.299)	(0.342)		(0.351)	(0.324)	(0.353)
1000	13	0.103	0.119	0.115	0.114	0.115	0.115
		(0.299)	(0.323)	(0.335)	(0.335)	(0.335)	(0.335)
	30	0.101	0.119	――	0.100	0.105	0.114
		(0.299)	(0.334)		(0.330)	(0.333)	(0.338)
	50	0.119	0.128	――	0.099	0.122	0.123
		(0.299)	(0.338)		(0.329)	(0.333)	(0.339)
	80	0.137	0.140	――	0.097	0.138	0.141
		(0.299)	(0.332)		(0.332)	(0.331)	(0.335)
2000	13	0.093	0.103	0.095	0.095	0.095	0.095
		(.299)	(0.313)	(0.326)	(0.324)	(0.326)	(0.326)
	30	0.096	0.109	――	0.094	0.096	0.100
		(0.299)	(0.319)		(0.325)	(0.321)	(0.336)
	50	0.097	0.106	――	0.089	0.099	0.103
		(0.299)	(0.321)		(0.328)	(0.320)	(0.324)
	80	0.108	0.110	――	0.093	0.106	0.105
		(0.299)	(0.312)		(0.326)	(0.318)	(0.322)

2 種類の提案手法はどちらも Restricted CAT や Big M のみでテストを構成するよりも能力推定精度が改善できている．この結果から，露出率を制限した場合にもテスト序盤には，MFI よりも MEPV などの能力推定誤差を考慮した項目選択基準を用いることが有効であると考えられる．Restriced CAT は，アイテムバンクサイズが比較的小さい場合（$N = 100, 500$）において，他手法に比べて能力推定精度が劣っていた．一方で，露出率の最大値を見ると設定した最大値 r_{max} を超過しなかった手法は Restricted CAT のみであった．Big M 法は最大露出率を超えた項目にペナルティを課すが，適した項目が不足した場合には最大露出率を超えた項目も出題可能であるため，r_{max} を超過している．Tree-CAT や Merged Tree-CAT は，決定木の各分枝における受検者の到達確率によって，露出率を制限する．この到達確率は，親節点への到達確率と親節点での回答確率，推定能力値の事後分布を用いて計算されるが，この値は推定能力値の事前分布の影響を受ける．したがって，事前分布と実際の受検者の能力値の分布に差異がある場合，決定木構築時に想定していた到達確率と実際の到達確率にも差異が生じる可能性がある．今回の実験では，決定木構築時の事前分布が $\theta\ N(0,1)$ に対し，受検者の真の能力値が $\theta\ U(-4,4)$ であったことが，設定した最大露出率 r_{max} を超過した要因であると考えられる．それに対し，Restricted CAT は項目選択の度に露出率を計算し，設定した最大露出率を超える項目はアイテムバンクから完全に除外されるため，最大露出率は超過していない．決定木から Restricted CAT へ切り替えることで，決定木による項目選択時に設定した最大露出率を超過した場合でも，切り替え後は選択不可能になるため，決定木のみでテスト構成するよりも露出率の最大値の超過を抑制できる．

Merged Tree-CAT と比較すると，決定木から Big$M$ 法へ切り替える提案手法は能力推定精度を改善できていた．しかし，アイテムバンクサイズが 500 や 1000 の場合は，Merged Tree-CAT の方が決定木から Restricted CAT へ切り替える提案手法よりも能力推定精度が高かった．これは，切り替え後の Restricted CAT の能力推定精度の影響を受けていると思われる．Restricted

CAT は上述した様に，他の手法と異なり，設定した最大露出率を超える項目はアイテムバンクから完全に除外され選択不可能となる．そのため，アイテムバンクサイズが小さい場合は，露出可能かつ質が高い項目の数が不足しやすくなることから，能力推定精度が他手法に比べ低くなる．決定木から Restricted CAT へ切り替える提案手法もこの影響を受けるため，アイテムバンクサイズが 500 の場合の様に小さい場合には，Merged Tree-CAT よりも能力推定精度が低くなったと考えられる．アイテムバンクサイズが 1000 や 2000 の様に大きい場合には，決定木から Restricted CAT へ切り替える提案手法も Big M 法へ切り替える提案手法と同様に Merged Tree-CAT よりも能力推定精度の改善が確認できた．これは，露出率を制限しない場合と同様に，提案手法では分枝統合を行わないため，Merged Tree-CAT の分枝統合の度に発生する能力推定誤差の問題や分枝統合を繰り返すことで出題不可能な項目が増加する問題の影響を受けないためである．

能力推定誤差に関しては，決定木から Big$M$ 法へ切り替える提案手法が最も小さい結果となったが，平均露出率を見ると Restricted CAT や Merged Tree-CAT の方が小さく項目露出の偏りが小さかった．Restricted CAT の平均露出率が小さいのは，最大露出率を厳密に制限しているためである．Merged Tree-CAT の平均露出率が小さいのは，分枝統合の際に出題済みの項目群を 3.2.2 節の式 (13) のように統合する．統合される分枝同士の出題済みの項目群が完全に一致している場合を除き，出題済みの項目群（すなわち，出題不可能な項目群）が増加する．出題項目数が多くなるほど分枝統合の回数も増えるため，この影響を強く受ける．そのため，出題項目数が多くなるほど Merged Tree-CAT の平均露出率は提案手法や Big M に比べて小さくなっている．

続いて，実データからなるアイテムバンクを用いて同様の実験を行った．表 8 は各手法の推定能力値の RMSE を表し，表 9 は各手法の露出率の平均値と最大値を表す．

表 8　実データを用いて露出率を制約とした各手法における RMSEE

アイテムバンク	出題項目数	RCAT (MFI)	BigM (MFI)	Tree (MEPV)	M Tree (MEPV)	提案 RCAT (MEPV,MFI)	提案 BigM (MEPV,MFI)
978	13	0.605	0.561	0.542	0.545	0.542	0.542
	30	0.413	0.381	—	0.377	0.395	0.372
	50	0.345	0.338	—	0.333	0.334	0.324
	80	0.282	0.280	—	0.294	0.275	0.272

表 9　実データを用いて露出率を制約とした各手法における露出率の平均値（最大値）

アイテムバンク	出題項目数	RCAT (MFI)	BigM (MFI)	Tree (MEPV)	M Tree (MEPV)	提案 RCAT (MEPV,MFI)	提案 BigM (MEPV,MFI)
978	13	0.125	0.139	0.128	0.126	0.128	0.128
		(0.299)	(0.346)	(0.354)	(0.352)	(0.354)	(0.354)
	30	0.124	0.142	—	0.117	0.120	0.135
		(0.299)	(0.342)		(0.341)	(0.329)	(0.347)
	50	0.131	0.144	—	0.115	0.126	0.138
		(0.299)	(0.337)		(0.338)	(0.324)	(0.343)
	80	0.140	0.147	—	0.119	0.133	0.142
		(0.299)	(0.326)		(0.342)	(0.319)	(0.330)

露出率を制限しない場合の結果と同じ理由で，シミュレーションデータよりも全体的に能力推定精度が比較的低く，露出率の平均値と最大値はシミュレーションデータを用いた場合よりも比較的高くなっていた．実データにおいても提案手法は，Restricted CAT や Big M のみを使用した場合より

も，能力推定精度が改善できた．Merged Tree-CAT と比較すると，決定木から Big M へ切り替える提案手法は能力推定精度が改善できたが，決定木から Restricted CAT へ切り替える提案手法は，一部の条件で能力推定精度が劣っていた．これは，識別力パラメータの値にばらつきがある今回の実データの場合，Restricted CAT の様に厳密に r_{max} を超えないように項目選択する露出制御手法で，識別力パラメータの高い項目の露出が偏り易い MFI を使用したとき，能力推定誤差が小さくなったテスト後半に露出可能かつ識別力パラメータが高い項目の数が不足してしまい易いことが原因であると考えられる．その他の点は，シミュレーションデータにおける結果と概ね同じであった．

6　むすび

本論文では，決定木を用いた適応型テストから通常の適応型テストへ切り替える手法を提案した．実験結果から，決定木の分枝数の増加に伴う時間・空間計算量の増加によって構成可能な決定木の大きさが限定的である問題を改善できた．また，分枝統合を行いながら決定木を構築する Merged Tree-CAT に比べ，提案手法は能力推定精度を改善できることがわかった．これは分枝統合を行った際に生じる推定能力値の誤差と，出題不可能な項目群が増加することにより情報量の高い項目が出題できなくなるためである．出題項目数が多くなるほど分枝統合の回数が増えるため，能力推定精度の低下がより顕著になる．そのため能力推定精度の観点から見た場合，分枝統合をせずに可能な限り大きな決定木を構成し，その後一般的な適応型テストへ切り替える手法の方が適していると考えられる．また，テスト序盤に MEPV の様な能力推定誤差を考慮できる項目選択基準を用いて項目選択し，能力推定誤差がある程度小さくなった状態でフィッシャー情報量を用いた項目選択基準へ切り替えることで，フィッシャー情報量のみを用いた項目選択した場合よりも能力推定精度を改善できた．

しかし，露出率を制限した場合の提案手法は Merged Tree-CAT や，Restricted CAT と比べて項目露出の偏り（平均露出率）が大きい傾向が見られた．また，各手法に対して同じ制約（露出率の最大値）を課して実験した場合でも，露出率の最大値を厳格に制御する Restricted CAT に切り替えた手法では，項目露出の偏りは減少するが，能力推定誤差は増加する傾向があり，Big M に切り替えた手法では，能力推定誤差は減少するが，項目露出の偏りは増加する傾向があった．3.3.6 節で述べた様に，項目露出の減少と能力推定誤差の増加はトレードオフの関係にあるため，平均露出率を揃えた状態で能力推定誤差を比較することが望ましいが，平均露出率は全受検者に対してテストの実施が完了するまで分からないため，正確に揃えることは困難である．今後は，露出率の制約に関してパラメータチューニングを行い，平均露出率を揃えた状態で能力推定誤差を比較することや，3.3.6 節で紹介した情報量を等質にする項目選択手法を用いて能力推定誤差を揃えた状態で露出率を比較することを検討する．また，3.3 章で紹介した項目露出制御手法 [4–6, 11, 37–39, 42, 49, 51] を用いた決定木の構築や 2 段階適応型テストの実現についても検討する．

参考文献

[1] 植野真臣，永岡慶三，e テスティング，培風館，2009.

[2] M. Ueno, K. Fuchimoto, and E. Tsutumi, “e-testing from artificial intelligence approach,” Behaviormetrika, vol.48, no.2, pp.409–424, 2021.

[3] M. Ueno, “Ai based e-testing as a common yardstick for measuring human abilities,”The 18th International Joint Conference on Computer Science and Software engineering, pp.1–6, IEEE computer society, 2021.

[4] W. van der Linden and C. Glas, eds., Elements of Adaptive Testing (Statistics for Social and Behavioral Sciences), Springer, 2010.

[5] W. van der Linden and B.P. Veldkamp, “Constraining item exposure in computerized adaptive testing with shadow tests,”J. Educational and Behavioral Statistics, vol.29, no.3, pp.273–291, 2004.

[6] W. van der Linden and S. Choi, “Improving item-exposure control in adaptive testing,”J. Educational Measurement, vol.57, no.3, pp.405–422, 2019.

[7] H. -H. Chang and Z. Ying, ”A global information approach to computerized adaptive testing,” Applied Psychological Measurement, 20, 213–229, 1996.

[8] W. J. J. Veerkamp and M. P. F. Berger, ”Item-selection criteria for adaptive testing,” Journal of Educational and Behavioral Statistics, 22, 203–226, 1997.

[9] Veerkamp, W. J. J. (1996). Statistical inference for adaptive testing (Internal report). Enschede, The Netherlands: University of Twente, Department of Educational Measurement and Data Analysis.

[10] van der Linden, W.J. (1998) Bayesian item selection criteria for adaptive testing. Psychometrika 63, 201–216.

[11] W. van der Linden and L.M. Reese, “A model for optimal constrained adaptive testing,” Applied Psychological Measurement,vol.22, no.3, pp.259–270, 1998.

[12] M. Ueno, P. Songmuang (2010), “Computerized Adaptive Testing based on Decision Tree”, The 10th IEEE International Conference on Advanced Learning Technologies, pp.191-193.

[13] M. Ueno (2013), “Adaptive Testing Based on Bayesian Decision Theory”, Artificial Intelligence in Education 2013, pp.712-716.

[14] Raiffa, H., and Schlaifer, R. (1961) Applied Statistical Decision Theory, Harvard University Press.

[15] D. Delgado-Gomez, Juan C. Laria, Diego Ruiz-Hernandez (2019), “Computerized adaptive test and decision trees: A unifying approach”, Expert Systems With Applications 117 pp.358–366.

[16] Javier Rodríguez-Cuadrado, David Delgado-Gómez, Juan C. Laria, Sara Rodríguez-Cuadrado (2020), “Merged Tree-CAT: A fast method for building precise computerized adaptive tests based on decision trees”, Expert Systems With Applications 143 pp.113–120

[17] D. Yan and C. Lewis and M. Stocking (2004), ”Adaptive Testing With Regression Trees in the Presence of Multidimensionality”, Journal of Educational and Behavioral Statistics, No.3, Vol.29, pp.293–316.

[18] D. Delgado-Gomez, E. Baca-Garcia, D. Aguado, P. Courtet, J. LopezCastroman (2016), “Computerized Adaptive Test vs. decision trees: Development of a support decision system to identify suicidal behavior”, Journal of Affective Disorders 206 pp.204-209.

[19] Samejima, F. (2016). Graded response models. In Handbook of Item Response Theory, Volume One, pages 123–136. Chapman and Hall/CRC

[20] van der Linden, W. J. and Glas, C. A. (2000). Computerized adaptive testing: Theory and practice. Springer

[21] van der Linden, W. J. and Pashley, P. J. (2009). Item selection and ability estimation in adaptive testing. In Elements of adaptive testing, pages 3–30. Springer. doi:10.1007/978-0-387-85461-8 1.

[22] van der Linden, W. J. and Veldkamp, B. P. (2005). Constraining item exposure in computerized adaptive testing with shadow tests, volume 2. Law School Admission Council.

[23] van der Linden, W. J. and Veldkamp, B. P. (2007). Conditional itemexposure control in adaptive testing using item-ineligibility probabili622 ties. Journal of Educational and Behavioral Statistics, 32(4):398–418. doi:10.3102/1076998606298044.

[24] Sympson, J. and Hetter, R. (1985). Controlling item-exposure rates in computerized adaptive testing. In Proceedings of the 27th annual meeting of the Military Testing Association, pages 973–977.

[25] Chang, H.-H. and Ying, Z. (1996). A global information approach to computerized adaptive testing. Applied Psychological Measurement, 20(3):213–229. doi:10.1177/014662169602000303

[26] Lord, F. M. (1980). Applications of item response theory to practical testing problems. Hillsdale, NJ: Lawrence Erlbaum.

[27] A. Birnbaum, "Some latent trait models and their use in inferring an examinee's ability," Statistical Theories of Mental Test Scores, eds. by F.M. Lord and M.R. Novick, pp.397–479, Addison-Wesley, 1968

[28] F.B. Baker and S.-H. Kim, eds., Item Response Theory: Parameter Estimation Techniques, CRC Press, July 2004.

[29] W. van der Linden, ed., Handbook of Item Response Theory, Volume One: Models, Chapman and Hall/CRC, 2016.

[30] W. van der Linden, ed., Handbook of Item Response Theory, Volume Two: Statistical Tools, Chapman and Hall/CRC, 2016.

[31] R.D. Bock and R.J. Mislevy, "Adaptive eap estimation of ability in a mi-

crocomputer environment,” Applied Psychological Measurement, vol.6, no.4, pp.431–444, 1982.

[32] W.D. Way,“Protecting the integrity of computerized testing item pools,” Educational Measurement: Issues and Practice, vol.17, pp.17–27, 1998.

[33] H. Wainer, “Cats: Whither and whence,” Psicológica, vol.21, no.1, pp.121–133, 2000

[34] M. Ueno and T. Okamoto, “Item response theory for peer assessment,” Proc. IEEE International Conference on Advanced Learning Technologies, pp.554–558, 2008.

[35] M. Uto and M. Ueno,“Item response theory for peer assessment,”IEEE Transactions on Learning Technologies, vol.9, no.2, pp.157–170, 2016

[36] 宇都雅輝，植野真臣，“ピアアセスメントの低次評価者母数をもつ項目反応理論,”電子情報通信学会論文誌.D，vol.98，no.1，pp.3–16，2015.

[37] Y. Miyazawa and M. Ueno,“Computerized adaptive testing method using integer programming to minimize item exposure,”Advances in Artificial Intelligence: Selected Papers from the Annual Conference of Japanese Society of Artificial Intelligence (JSAI 2019), pp.105–113, Springer, 2020.

[38] 宮澤芳光, 宇都雅輝, 石井隆稔, 植野真臣,“測定精度の偏り軽減のための等質適応型テストの提案,”信学論（D），vol.J101-D, no.6, pp.909–920, June 2018.

[39] M. Ueno and Y. Miyazawa, “Uniform adaptive testing using maximum clique algorithm,” 20th International Conference, AIED 2019, pp.482–493, 2019

[40] 宮澤芳光，植野真臣，”高精度能力推定を保証する 2 段階等質適応型テスト”電子情報通信学会論文誌 D Vol. J106–D No. 1 pp. 34–46 2022

[41] 石井隆稔，赤倉貴子，植野真臣,“複数等質テスト構成における整数計画問題を用いた最大クリーク探索の近似法,”信学論（D），vol.J100-D,

no.1, pp.47–59, Jan. 2017

[42] van der Linden WJ, Choi SW (2019) Improving item-exposure control in adaptive testing. J Educ Meas.

[43] S.W. Choi, S. Lim, and W. van der Linden, "Testdesign: an optimal test design approach to constructing fixed and adaptive tests in r," Behaviormetrika, vol.49, pp.191–229, 2022.

[44] S.W. Choi and S. Lim, "Adaptive test assembly with a mix of setbased and discrete items," Behaviormetrika, vol.49, pp.231–254, 2022.

[45] J. Mislevy, R, "Bayes modal estimation in item response models," Psychometrika, vol.51, no.2, pp.177–195, 1986.

[46] Weiss, D. J. (1982). Improving measurement quality and efficiency with adaptive testing. Applied psychological measurement, 6(4):473–492. doi:10.1177/014662168200600408.

[47] Veerkamp, W. J. and Berger, M. P. (1997). Some new item selection criteria for adaptive testing. Journal of Educational and Behavioral Statistics, 22(2):203–226. doi:10.3102/10769986022002203

[48] Weissman, A. (2007). Mutual information item selection in adaptive classification testing. Educational and Psychological Measurement, 67(1):41–58. doi:10.1177/0013164406288164.

[49] Kingsbury, G. G. and Zara, A. R. (1989). Procedures for selecting items for computerized adaptive tests. Applied measurement in education, 2(4):359–375. doi:10.1207/s15324818ame0204 6.

[50] van der Linden, W. J. (2003). Some alternatives to sympson-hetter item exposure control in computerized adaptive testing. Journal of Educational and Behavioral Statistics, 28(3):249–265. doi:10.3102/10769986028003249.

[51] Revuelta, J. and Ponsoda, V. (1998). A comparison of item exposure control methods in computerized adaptive testing. Journal

of Educational Measure580 ment, 35(4):311–327. doi:10.1111/j.1745-3984.1998.tb00541.

[52] Raiffa, H., Schlaifer, R.: Applied Statistical Decision Theory. Harvard Business School Publications (1961)

[53] Recruit, “Synthetic personality inventory(SPI),” http://www.spi.recruit.co.jp/

[54] Williams, H. P. (1990). Model building in mathematical programming. New York, NY: Wiley.