統計数理(2024)
第 72 巻 第 1 号 43–59

特集「心理統計学の新展開」
[原著論文]

項目露出ペナルティを用いた整数計画法による自動並行テスト構成

渕本 壱真[†]・植野 真臣[†]

(受付 2023 年 6 月 26 日；改訂 12 月 27 日；採択 2024 年 1 月 24 日)

要　　旨

e-Testing の特徴は異なる問題で構成されるが同一精度の測定を実現できるテストの自動構成であり，その重要な課題は可能な限り多くのテストを生成することである．自動テスト構成手法は多数存在するが，整数計画法を用いた最大クリークが現在最も多くのテストを高い測定精度で生成できることが報告されている．しかし，この手法は，テスト間に項目の重複を許すため，項目の出題頻度に偏りを生じさせ，テストの信頼性を低下させる．この問題を解決するために，本研究では整数計画法の目的関数に露出数を所与としたロジスティック関数による以下の 2 種類のペナルティ，(1)ロジスティック関数による決定論的ペナルティ，(2)ロジスティック関数による確率論的ペナルティ，を提案する．数値実験により，提案手法はテスト数を減らすことなく露出数の偏りを減らすことを示す．

キーワード：自動テスト構成，項目反応理論，e-Testing，項目露出問題，整数計画法．

1. はじめに

e-Testing とは，異なる問題で構成されるが，同一精度の測定を実現できるコンピュータテストのことである（Ueno, 2021; Ueno et al., 2021）．e-Testing を用いることで，同一能力の受検者が異なるテストを受検しても同一得点となる保証がある．そのために，受検者が同一精度で複数回受検が可能となる．我が国においても医療系大学間共用試験や情報処理技術者試験などが e-Testing で行われている．また，大学入学試験や公務員試験での導入も検討されている（植野, 2023）．

歴史的には，e-Testing のアイデアは Lord and Novick (1968)が同じ真の得点を測定する 2 つのテストについて並行テストと定義したことから始まる．しかし，並行テストは古典的テスト理論における仮定であり，このようなテストの実現は困難であった．そのため，Samejima (1977)は項目反応理論(Item Response Theory: IRT)（例えば，van der Linden, 2017)を用いて，並行テストの概念を拡張した．具体的には，項目反応理論におけるテスト情報量の逆数が受検者能力推定値の漸近分散に収束することを用いて，この値が等価なテストを弱並行テストとして定義した．e-Testing の普及に伴い，この弱並行テストの概念に基づいた自動テスト構成手法が数多く提案されている（Boekkooi-Timminga, 1990; Armstrong et al., 1994, 1998; van der Linden and Adema, 1998; van der Linden, 2005; Songmuang and Ueno, 2011; Ishii et al., 2013, 2014; Ishii and Ueno, 2017; Fuchimoto et al., 2022）．

[†] 電気通信大学 大学院情報理工学研究科：〒 182–8585 東京都調布市調布ヶ丘 1–5–1

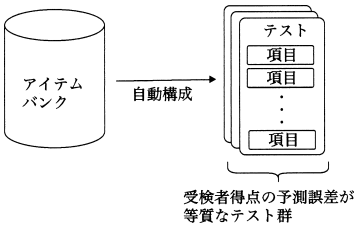

図 1. アイテムバンクからの自動テスト構成.

一般に，e-Testing ではテストの管理方法としてアイテムバンク方式が用いられる．アイテムバンクとは出題する問題(以降，項目と呼ぶ)の出題分野や統計データ等を格納しているデータベースのことである．自動テスト構成では所望のテストの性質を満たす項目の組合せをアイテムバンクから計算機により探索する．

図 1 は自動テスト構成手法の概念図である．一般に，自動テスト構成ではアイテムバンクから互いに受検者得点の予測測定誤差が等質となるように異なる項目の組合せを列挙する．これにより，同一能力の受検者が異なるテストを受けても同一の得点となることが保証される．

自動テスト構成手法で最も有名な Big Shadow Test method(BST 法)では混合整数計画法を用いて，目標とする受検者の測定誤差からの差異が最小となるテストを貪欲法で逐次的に構成する（van der Linden, 2005)．しかし，BST 法は逐次的に誤差が小さいテストから構成するため，テスト数の増加につれて，受検者の予測測定誤差が大きくなる問題がある．そのため，BST 法はテスト数を十分に確保できず，実用的でない．例えば，年間 20 万人以上受検する情報処理技術者試験の一区分「IT パスポート試験」(独立行政法人情報処理推進機構, 2023)では全国 47 都道府県に設置された複数の会場で月に数回開催されている．また，年間 1 万人以上受検する医療系大学間共用試験（公益社団法人医療系大学間共用試験実施評価機構, 2023)では各大学ごとにカリキュラム(臨床実習開始時期)に応じて異なる時期にテストを受検する必要がある．さらに，これらの試験ではカンニング等の不正防止のために，同一試験会場であっても受検者ごとに異なるテストを出題している．したがって，受検者数以上のテストを用意しなければ同じ項目で構成されるテストが再出題され，テストの信頼性低下につながる（Wainer, 2000)．

この問題を解決するために，Ishii et al. (2013)はその当時，最も多くのテストを構成する最大クリーク法を提案した．この手法は自動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させる．具体的には，与えられたアイテムバンク・テスト構成条件で構成可能な全てのテストを頂点集合とし 2 つのテストが等質かつ共通する項目の数が一定数以下である場合に，頂点(テスト)間に辺を引いたグラフからクリークと呼ばれる任意の 2 頂点が隣接している最大の部分グラフ構造を探索することで自動テスト構成を行う．

この手法は理論的に最大数のテスト構成を保証するが，構成可能な全てのテストを頂点とするグラフ構造は組合せ爆発的に大きくなるため，最大クリークを探索することやグラフ構造をメモリ上に保存することは困難である．そのため，Ishii et al. (2014)はグラフ全域から部分グラフをランダムに抽出し，ここから最大クリーク探索を繰り返すことによりグラフ全体の最大クリークを近似的に探索する手法を提案した．本手法により，当時の既存研究よりも 10〜100 倍以上多くのテストを構成できるようになっている．

しかし，最大クリーク探索は最先端の最大クリーク探索手法（Tomita et al., 2016; Li et al., 2017)を用いても，$O(|V|^2)$($|V|$ はグラフの頂点数)の空間計算量を少なくとも必要とするため，(著者らの計算機環境で)最大で 10 万のテストを構成することが限界であった．そこで，Ishii

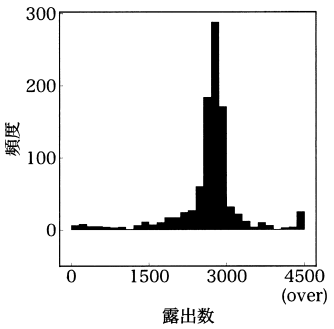

図 2. 露出数のヒストグラムの例.

and Ueno (2017)は探索中のクリークの全頂点と隣接する頂点を整数計画法を用いて逐次的に探索することで，計算に必要な空間計算量を $O(|V|)$ へ減少させる手法を提案した．これにより，10 万を超えるテストを構成できるようにした．ただし，整数計画法の時間計算量が $O(2^n)$ (n はアイテムバンクの項目数)と大きく，テスト構成数の改善は僅かなものであった．

整数計画法の計算時間を改善するために，Fuchimoto et al. (2022)は探索中のクリークの全頂点と隣接する頂点を並列探索する Hybrid Maximum Clique Algorithm Using Parallel Integer Programming method(HMCAPIP 法)を提案した．本手法により，最も時間を要した整数計画法による頂点探索を並列化することで探索時間を大幅に減少できた．具体的には多くの条件下で探索時間を非並列化時の約 25% 程度に抑えられた (Fuchimoto et al., 2022)．さらに，並列化探索で得られた頂点を整数計画法の目的関数の下限値として用いることで分枝限定法の効果により探索を高速化した．結果として，最大 7 日間で当時の最新研究の約 2.7 倍にあたる約 27 万のテスト生成を実現している (Fuchimoto et al., 2022)．

しかし，これらの手法はテスト間に重複を許すことで各項目の出題回数(以降，露出数と呼ぶ)に偏りが生じる．図 2 は Synthetic Personality Inventory (SPI)試験 (Recruit, 2023)で実際に用いられていたアイテムバンクを用いて従来手法 (Fuchimoto et al., 2022)によりテスト構成したときの各項目の露出数のヒストグラムを示したものである．図 2 より，露出数のばらつきが大きいことがわかる．特に，4500 回以上出題される項目やほとんど出題されない項目が存在する．このような露出数の偏りはさまざまな弊害を生じる．最も深刻な問題は，露出数の大きい項目が受検者間で共有され受検対策されてしまい，その項目の信頼性を低下させることである (Wainer, 2000)．また，作問にかかるコストは多大であり，露出の低い項目は可能な限り出題することが望ましい．そのため，露出数は可能な限り一様であることが理想的である．

露出数の偏りを軽減するために，Ishii and Ueno (2015)は最大クリーク法と整数計画法を用いた手法によりテストを構成し，その中から最も露出率が小さいテスト構成を選択するアルゴリズムを提案した．具体的には，探索した全てのテスト群を候補として保存しておき，最後に候補中で最も露出率が小さいテスト群を出力する．これによって，従来手法よりも最大露出率を約 0.1%〜4% 軽減できた．しかし，これらの手法は露出の偏りを軽減する定式化がされていないため，その改善に限界がある．

その他に，適応型テストでは数多くの露出制御手法が提案されている (Hetter and Sympson, 1997; van der Linden and Reese, 1998; Stocking and Lewis, 1998, 2000; van der Linden and Veldkamp, 2004; van der Linden, 2017; van der Linden and Choi, 2020)．ここで，適応型テストとは受検者の能力を逐次的に推定しながら，その能力に応じて測定精度が最も高い項目を出題す

るテスト形式の一つである．最も代表的な手法として，van der Linden and Reese (1998)は最大露出数の上限を制約式に与えた整数計画問題を用いて項目を選択する手法を提案した．この手法は最大露出数を厳密に制御できるが，受検者の能力推定精度が低下する．この問題を緩和した手法として，確率的アプローチが提案されている（Hetter and Sympson, 1997; van der Linden and Reese, 1998; Stocking and Lewis, 1998, 2000; van der Linden and Veldkamp, 2004; van der Linden, 2017; van der Linden and Choi, 2020）．例えば，Sympson-Hetter 法では各項目の露出数に応じた出題確率をシミュレーション実験により求め，その確率に応じて出題を行う（Hetter and Sympson, 1997; Stocking and Lewis, 1998, 2000）．さらに，シミュレーション実験不要な手法として，van der Linden らは各項目の出題可能な確率を意味する適格確率を定義し，受検者ごとに適格確率に従ってアイテムバンクからその項目を除くことで露出数を制御する手法を提案した（van der Linden and Reese, 1998; van der Linden and Veldkamp, 2004; van der Linden, 2017; van der Linden and Choi, 2020）．しかし，これらの手法（Hetter and Sympson, 1997; van der Linden and Reese, 1998; Stocking and Lewis, 1998, 2000; van der Linden and Veldkamp, 2004; van der Linden, 2017; van der Linden and Choi, 2020)は最大露出率の抑制のみを目的としており，露出数を一様に近づけることは限定的である．そのため，これらのアイデアを並行テストに適用することは可能かもしれないが，本論文が目標としている露出数を一様に近づけ各項目の露出数を減少させることは難しい．

露出数の偏りを防ぐために，本研究では露出数を所与としたロジスティック関数による 2 種類のペナルティ項を HMCAPIP 法における整数計画問題の目的関数に追加することを提案する．1 つ目はこのロジスティック関数を用いた決定論的ペナルティ項である．この決定論的ペナルティ項は露出数に応じた負の重みを常に各項目の決定変数に与える．2 つ目はロジスティック関数を用いた確率論的ペナルティ項である．確率論的ペナルティ項は数理計画法の Big-M 法（Williams, 1990)に基づいて，露出数に応じた確率により，大きな負の重みを各項目の決定変数に与える．

本論文では，提案手法の有効性をシミュレーション及び実データを用いて示した．具体的には，従来手法と比較して，テスト構成数を減少させることなく，露出数の偏りを抑制できることを示した．

2. 項目反応理論

項目反応理論は受検者の項目への正答確率をモデル化し，異なる項目から構成されるテストを受けた受検者の能力を同一尺度上で評価できる．

一般的に，テストでは項目 $i(=1,\ldots,n)$ に対する受検者 $j(=1,\ldots,m)$ の反応 u_{ij} を以下のように表す．

$$u_{i,j}=\begin{cases}1 & i \text{ 番目の項目に受検者 } j \text{ が正答}\\ 0 & \text{それ以外}\end{cases}$$

本論文では，項目反応理論の中で最もよく使われている 2 母数ロジスティックモデル(2-Parameter Logistic Model:2PLM)を用いる．このモデルでは，能力値 $\theta_j \in (-\infty,\infty)$ を持つ受検者 j が項目 i に正答する確率 $p_i(\theta_j)$ を以下のように定義する．

$$\begin{aligned} p_i(\theta_j) &\equiv p(u_{ij}=1|\theta_j) \\ &= \frac{1}{1+\exp(-1.7a_i(\theta_j-b_i))} \end{aligned} \tag{2.1}$$

ここで，$a_i \in [0,\infty), b_i \in (-\infty,\infty)$ はそれぞれ i 番目の項目の識別力パラメータ，難易度パ

ラメータと呼ばれる項目パラメータである．

IRT では項目 i において，式(2.1)を用いて計算したフィッシャー情報量を項目情報量 $I_i(\theta)$ (Item Information)と呼び，以下のように表す．

$$I_i(\theta) = 1.7^2 a_i^2 p_i(\theta)(1 - p_i(\theta)) \tag{2.2}$$

また，テストに含まれる項目の項目情報量の総和をテスト情報量と呼び，以下のように表す．

$$I(\theta) = \sum_{i \in T} I_i(\theta) \tag{2.3}$$

ここで，T はテストに含まれる項目の集合である．このテスト情報量の逆数が受検者能力推定値の漸近分散に収束する（Lord and Novick, 1968）．Samejima（1977)は，このテスト情報量が等価なテストを弱並行テストとして定義した．ただし，この時点の弱並行テストはテスト項目数やカテゴリ区分などの制約を一切課していない．多くの自動テスト構成手法（Boekkooi-Timminga, 1990; Armstrong et al., 1994, 1998; van der Linden and Adema, 1998; van der Linden, 2005; Songmuang and Ueno, 2011; Ishii et al., 2013, 2014; Ishii and Ueno, 2017; Fuchimoto et al., 2022）がこの弱並行テストの定義に基づいている．

全ての受検者の能力値 $\theta_j \in (-\infty, \infty)$ について，テスト情報量が等価なテストを生成するのは困難である．そのため，実際にはテスト情報量における受検者の能力値 θ_k を $\theta_k = \{\theta_1, \theta_2, \ldots, \theta_K\}$ のように幾つかの点でサンプリングし，離散的に扱う．例えば，van der Linden（2005)は混合整数計画問題を用いて，サンプリングした各点 θ_k ごとにテスト情報量の目標値を設定し，その目標値との誤差の和が最小となるテスト情報量を持つテストを逐次的に構成している．しかし，この手法は誤差が小さいテストから逐次的に構成するため，テスト数が増加するに従って，受検者の予測測定誤差が大きくなる問題がある．

この問題を解決するために，Ishii et al.（2014)は θ_k におけるテスト情報量の上限・下限制約（UB(θ_k)，LB(θ_k))を設定し，全ての制約を満たすテストを受検者得点の予測測定誤差が等質であるとした．これにより，受検者の予測測定誤差の許容範囲を事前に設定できるため，テスト数が増加するに従って，受検者の予測測定誤差が大きくなる問題が解消される．図 3 は，表 1 に示した並行テストのテスト情報量への上限下限の例である．図中の #1〜#4 は構成テストの情報量関数である．#1, #2 は共に制約を満たしており等質である．一方で，#3, #4 は制約を満たしておらず等質でない．

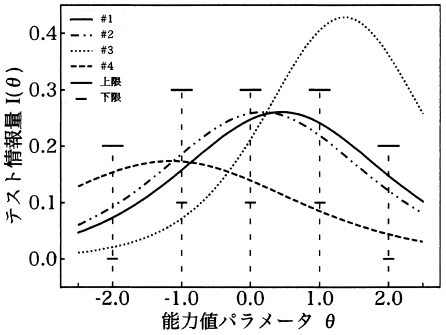

図 3. テスト情報量への上限下限の例.

表 1. テスト情報量制約の例.

テスト情報量 $I(\theta)$（下限値/上限値）				
$\theta = -2.0$	$\theta = -1.0$	$\theta = 0.0$	$\theta = 1.0$	$\theta = 2.0$
0.0/0.2	0.1/0.3	0.1/0.3	0.1/0.3	0.0/0.2

3. 自動テスト構成アルゴリズム

本章では，現在最も多くのテストを生成可能な最大クリーク法を用いた自動テスト構成手法を紹介する．

3.1 最大クリーク法を用いた自動テスト構成

Ishii et al. (2013)は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とで，厳密に最大数のテストを構成する手法を提案した．ここで，クリークとは任意の 2 頂点が隣接している部分グラフである．具体的には，無向グラフ $G = (V, E)$ を頂点の有限集合 V と辺の集合を E としたとき，最大クリーク法は次のように定式化できる．

$$\textbf{variables} \quad C \subseteq V \tag{3.1}$$

$$\textbf{maximize} \quad |C| \tag{3.2}$$

$$\textbf{subject to} \quad \forall v, \forall w \in C, \{v, w\} \in E \tag{3.3}$$

$\{v, w\} \in E$ は頂点 v, w に辺が引かれていることを意味する．

最大クリーク法によるテスト構成は図 4 のように，テスト候補を以下のグラフ構造とみなし，その中から最大クリーク探索を行うことで，テスト構成する．

（頂点） 与えられたアイテムバンクからテストの構成条件を満たす全てのテストを頂点とする．

（辺） 2 つのテスト候補の共通する項目数が一定値以下（以降，OC と呼ぶ）の場合，その 2 つの頂点（テスト）間に辺を引く．

このグラフの任意の頂点はテスト構成条件を満たしている．さらに，クリーク中の任意の 2 頂点は隣接しており，OC を満たす．したがって，このクリーク中の頂点に対応するテストはそれぞれ等質であり，その中でも最大クリークは理論的に最大数を保証したテスト群となる．

最大クリーク法のアルゴリズムを以下に示す．ただし，詳細なアルゴリズムについては Ishii et al. (2013, 2014)を参照されたい．

(1) アイテムバンクの項目の全ての組合せから，OC 以外の条件を満たすテストの組合せを探索木（Ishii et al., 2013, 2014）を用いて全て列挙する．
(2) (1) で列挙したテストをそれぞれ頂点とみなし，2 つのテストの共通する項目数が OC 以下の場合，その頂点間に辺を引く．
(3) (1)，(2) で生成されたグラフから最大クリーク探索（例えば，Tomita et al., 2016 や Li et al., 2017）を行う．

この手法は厳密に最大数のテストを構成できるアルゴリズムであるが，時間，空間計算量が $O(2^{0.19171|V|})$，$O(|V|^2)$ と計算コストが高い（Nakanishi and Tomita, 2008）．特に，自動テスト構成ではグラフの頂点の総数 $|V|$ はアイテムバンクの項目数に対して，組合せ爆発的に増加する．そのため，実際のアイテムバンクは数千項目から構成されるため，最大クリーク探索を厳密に行うことが困難である．

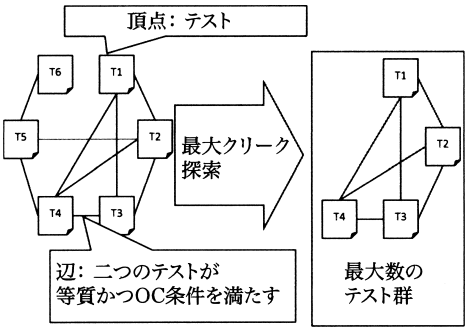

図 4. 最大クリーク法の概要図.

これらの計算コストを緩和するため，Ishii et al.（2014）は最大クリーク探索を行う近似アルゴリズム Random Maximum Clique Problem method（RndMCP 法）を提案した．最大クリーク法の問題点は，テスト構成数が増加するとグラフの探索空間が莫大となることである．そのため，RndMCP 法ではテスト候補グラフ全体から部分グラフをランダムに抽出し，このグラフから最大クリーク探索を繰り返す．これにより，グラフ全体の最大クリークを近似的に探索する．RndMCP 法は従来手法（van der Linden, 2005; Sun et al., 2008; Songmuang and Ueno, 2011）と比較して，10〜1000 倍以上多くのテストを構成できた．

3.2 整数計画法を用いた最大クリークアルゴリズム

RndMCP 法は空間計算量が大きく，10 万程度のテスト構成が上限であった．RndMCP 法の空間計算量を緩和するために，Fuchimoto et al.（2022）は整数計画法を用いた二段階並列探索手法である Hybrid Maximum Clique Algorithm Using Parallel Integer Programming method（HMCAPIP 法）を提案した．HMCAPIP 法の第一段階目は RndMCP 法でメモリが許す限り多くのテストを生成する．HMCAPIP 法の第二段階目は空間計算量が小さいが，時間計算量が大きい整数計画法による探索に切り替え，さらに多くのテストを生成する．具体的には，第一段階目に求めたクリークの全頂点と接続する頂点を以下の整数計画法により求める．

$$\textbf{maximize} \quad \sum_{i=1}^{n} \lambda_i x_i, \tag{3.4}$$

$$\textbf{subject to} \quad \sum_{i=1}^{n} = L, \tag{3.5}$$

$$\sum_{i=1}^{n} X_{i,t} x_i \leq \mathrm{OC} \qquad (t = 1, 2, \ldots, |C|), \tag{3.6}$$

$$LB_{\theta_k} \leq I(\theta_k) \leq UB_{\theta_k} \qquad (k = 1, 2, \ldots, K). \tag{3.7}$$

ここで，x_i は項目 i を選択する場合に 1，選択しない場合に 0 を示す決定変数，$X_{i,t}$ はクリーク中の t 番目のテストに項目 i が含まれる場合に 1, 含まれない場合に 0 を示す定数，L はテストの長さを表す定数である．また，λ_i は互いに独立な $[0, 1]$ の連続一様分布からの乱数であり，整数計画法を解く度にリサンプリングする．この乱数により，実行可能解の中からランダムにテストを生成できる．

さらに，Fuchimoto et al.（2022）は整数計画法の計算時間を緩和するために，頂点探索を並列化した．これにより，同一計算時間内において，HMCAPIP 法は最大 7 日間で当時の最新研

究の約 2.7 倍にあたる約 27 万のテスト生成可能としている（Fuchimoto et al., 2022）．

4. 提案手法

現在，HMCAPIP 法は世界で最も多くのテストを生成できる．しかし，HMCAPIP 法はテスト間に項目の重複を許すため各項目の露出数に偏りが生じる．この露出数の偏りはさまざまな弊害を生じる．最も深刻な問題は，露出数の大きい項目が受検者間で共有され受検対策されてしまい，その項目の信頼性を低下させることである（Wainer, 2000）．また，作問にかかるコストは多大であり，露出の低い項目は可能な限り出題することが望ましい．そのため，露出数は可能な限り一様であることが理想的である．

HMCAPIP 法においても，整数計画法の目的関数の決定変数に乱数の重み付け λ_i を与えることで，露出数の偏りを軽減している．しかし，各項目の特性(識別力パラメータや難易度パラメータ)に依存して，実行可能解中に既に露出数の偏りが生じており，λ_i による改善には限界がある．

この問題を緩和するために，本研究では既に生成したクリークの露出数を所与としたロジスティック関数によるペナルティを HMCAPIP 法における整数計画問題の目的関数に追加する．具体的には，以下の 2 種類のペナルティ，(1)ロジスティック関数による決定論的ペナルティ，(2)ロジスティック関数による確率論的ペナルティを提案し，露出数の標準偏差と最大露出率を従来手法よりも減少させる．

はじめに，現時点で生成したテスト群 U における，項目 i の露出数 IE_i を以下のように定義する．

$$IE_i = \sum_{t=1}^{|U|} X_{i,t}. \tag{4.1}$$

露出数 IE_i はテスト構成数に応じて増加する．そのため，提案手法ではテスト構成数に依らず露出数を共通尺度上で扱えるように，標準化露出数 z_i を以下のように定義する．

$$z_i = \frac{IE_i - IE_\mu}{IE_\sigma}. \tag{4.2}$$

ただし，IE_μ と IE_σ はそれぞれ全項目の露出数の平均値と標準偏差を示す．標準化露出数 z_i が標準正規分布に従うと仮定すると，その累積分布関数は次のロジスティック項目露出関数 $f(z_i)$ で近似できる．

$$f(z_i) = \frac{1}{1 + \exp^{-z_i}}. \tag{4.3}$$

ロジスティック項目露出関数 $f(z_i)$ は [0,1] の範囲における露出数の度合いとして解釈できる．また，ロジスティック項目露出関数 $f(z_i)$ は決定変数の重み λ_i の範囲に対応する．この性質を用いて，提案手法では HMCAPIP 法の目的関数に対する決定論的・確率論的ペナルティを 2 つ提案する．

4.1 決定論的ペナルティ

ロジスティック項目露出関数を $[0, 1]$ の度合いとして解釈すると，乱数の重み λ_i の値域と対応する．そのため，決定論的ペナルティではロジスティック項目露出関数を単純な決定変数のペナルティの重みとして扱うことを考える．具体的には以下の目的関数を最適化し，テストを生成する．本研究ではこの手法を提案手法(Det)と呼ぶ．

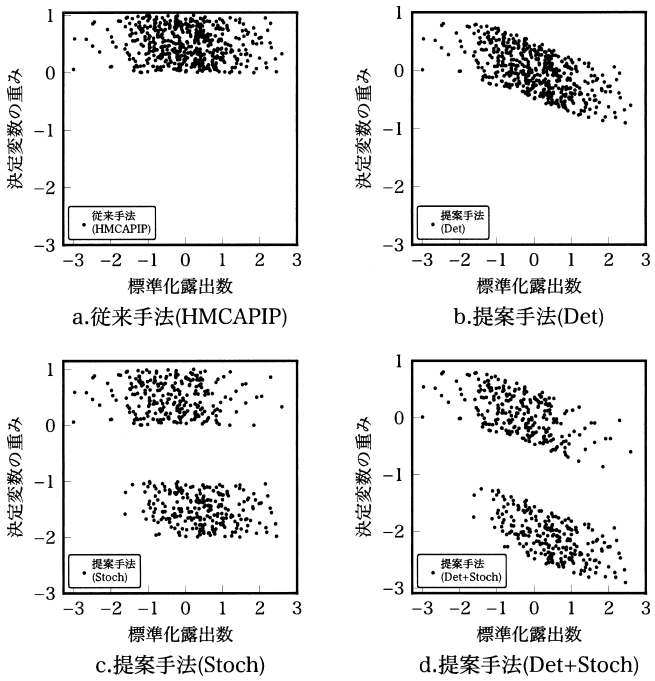

図 5. 標準化露出数と決定変数の重みの散布図.

$$\textbf{maximize} \qquad \sum_{i=1}^{n} \left(\lambda_i - \frac{1}{1+\exp^{-z_i}} \right) x_i. \tag{4.4}$$

数式(4.4)の意味を説明するために，図 5(a)及び図 5(b)に従来手法と提案手法(Det)における目的関数の決定変数の重み λ_i と $\lambda_i - \frac{1}{1+\exp^{-z_i}}$ をそれぞれ取り出し 500 個プロットした．ただし，標準化露出数は標準正規分布からの乱数，λ_i は互いに独立な $[0,1]$ の連続一様分布からの乱数としてデータを発生させた．図 5(a)より，従来手法では露出数を全く考慮しておらず偏りが生じる．一方で，図 5(b)より，提案手法(Det)では目的関数の決定変数の重みはロジスティック関数に従い，各項目の標準化露出数が小さい(大きい)ほど大きく(小さく)なる．これにより，この目的関数の最適解は現時点で露出数の高い項目の選択を抑制し，露出数の低い項目の選択を促進できる．したがって，提案手法(Det)では露出数の標準偏差を抑制したテスト構成が期待できる．

4.2 確率論的ペナルティ

提案手法(Det)では露出数の高い項目についても乱数の値によって，決定変数の重みが大きくなる．そのため，提案手法(Det)では露出数の高い項目に対するペナルティが小さい．本章では露出数の高い項目に対して，数理計画法の Big-M 法 (Williams, 1990)に基づいた重いペナルティを与えることを考える．Big-M 法は決定変数の重みの上界よりも大きなペナルティ M を与えることで，不必要な変数選択の個数を抑制できる定式化として知られている．例えば，提案手法において j 番目の決定変数 x_j の変数選択を可能な限り避けたいとする．このとき，以下の目的関数の値は x_j を含む実行可能解より含まない実行可能解の方が必ず大きな値を取る．したがって，決定変数 x_j を含む実行可能解は含まない実行可能解が存在しない場合を除き選

択されないことが保証される．

$$\textbf{maximize} \qquad \sum_{i=1}^{n} \lambda_i x_i - M x_j \qquad \left(\sum_{i=1}^{n} \lambda_i x_i < M \right). \tag{4.5}$$

この Big-M 法を用いて，確率論的ペナルティではロジスティック項目露出関数を確率とみなし，この確率に従ってペナルティ M を与える．具体的には以下の目的関数を最適化し，テストを生成する．本研究ではこの手法を提案手法(Stoch)と呼ぶ．

$$\textbf{maximize} \qquad \sum_{i=1}^{n} (\lambda_i - M_{i,p}) x_i, \tag{4.6}$$

$$M_{i,p} = \begin{cases} M & \eta_i < \frac{1}{1+\exp^{-z_i}}, \\ 0 & \text{otherwise.} \end{cases}$$

ここで，η_i は互いに独立な $[0,1]$ の連続一様分布からの乱数であり，整数計画法を解く度にリサンプリングする．

数式(4.6)の意味を説明するために，図 5(a)及び図 5(b)と同様の手順で，図 5(c)に従来手法と提案手法(Det)における目的関数の決定変数の重み $\lambda_i - M_{i,p}$ を取り出し 500 個プロットした．ただし，ペナルティの値は λ_i の最大値よりも大きい $M = 2$ とした．図 5(c)より，提案手法(Stoch)の決定変数の重みはペナルティが与えられない場合と与えられる場合でそれぞれ決定変数の値域が $[0,1]$ と $[0-M, 1-M]$ の 2 群に分離される．また，ペナルティ M が与えられる項目はロジスティック項目露出関数に従うため，標準化露出数が相対的に大きな(小さな)項目ほど，ペナルティが与えられる(与えられない)確率が高いことがわかる．結果として，この目的関数の最適解は極端に露出数の高い項目(最大露出数)の選択を避けることができる．一方で，提案手法(Stoch)では標準化露出数が極端に大きい項目のみを抑制するので，提案手法(Det)ほど露出数の標準偏差を抑えられない可能性がある．

4.3 決定論的・確率論的ペナルティ

提案手法(Det)及び提案手法(Stoch)はそれぞれ露出数の標準偏差と最大露出率を抑える効果が期待できる．これらの手法は独立して決定変数の重みにペナルティを与えられるため，単純な提案手法(Det)と提案手法(Stoch)の組合せにより，露出数の標準偏差と最大露出率を両方抑えることが可能かもしれない．具体的には以下の目的関数を最適化し，テストを生成する．この手法を提案手法(Det+Stoch)と呼ぶ．

$$\textbf{maximize} \qquad \sum_{i=1}^{n} \left(\lambda_i - \frac{1}{1+\exp^{-z_i}} - M_{i,p} \right) x_i, \tag{4.7}$$

$$M_{i,p} = \begin{cases} M & \eta_i < \frac{1}{1+\exp^{-z_i}}, \\ 0 & \text{otherwise.} \end{cases}$$

数式(4.7)の意味を説明するために，図 5(a)～(c)と同様の手順で，図 5(d)に提案手法(Det+Stoch)における目的関数の決定変数の重み $\lambda_i - \frac{1}{1+\exp^{-z_i}} - M_{i,p}$ を取り出し 500 個プロットした．提案手法(Det+Stoch)では各項目の露出数が小さい(大きい)ほど，大きく(小さく)なる．これにより，この目的関数の最適解は現時点で露出数の高い項目の選択を抑制し，露出数の低い項目の選択を促進できる．さらに，提案手法(Det+Stoch)ではペナルティ M を与えられると目的関数の決定変数の重みが極端に小さな値となり，標準化露出数の相対的に大きな項目ほど，ペナルティが与えられることがわかる．これにより，この目的関数の最適解は露出数

の高い項目ほど不必要な選択を避けることができる．

5. 評価実験

提案手法が従来手法よりもテスト構成数を減少させず，露出数の偏りを減少させることをシミュレーション実験により示す．具体的には従来手法(MCALIE 法 (Ishii and Ueno, 2015)及び HMCAPIP 法 (Fuchimoto et al., 2022))とテスト構成数，最大露出率及び露出数の標準偏差を比較した．ここで，最大露出率及び露出数の標準偏差は生成したテスト群 U における，項目 i の露出数 IE_i(式(4.1))と全項目の露出数 IE_i の平均値 IE_μ を用いて以下の通り計算した．

$$\text{最大露出率} = \frac{\max\{IE_1, IE_2, \ldots, IE_n\}}{|U|} \tag{5.1}$$

$$\text{露出数の標準偏差} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(IE_i - IE_\mu)^2} \tag{5.2}$$

また，各手法の計算時間は最大 24 時間とし，MCALIE 法と HMCAPIP 法のパラメータはそれぞれ，Ishii and Ueno (2015)と Fuchimoto et al. (2022)が実施した条件と同様である．アイテムバンクにはシミュレーション及び Synthetic Personality Inventory (SPI)試験 (Recruit, 2023) で実際に用いられていたデータを用いた．シミュレーションアイテムバンクは 2000 の項目を持ち，各項目の識別力パラメータを $a \sim U(0,1)$，難易度パラメータを $b \sim N(0,1^2)$ として発生させた．実データアイテムバンクの詳細は表 2 の通りである．

本研究では，これらのアイテムバンクから表 3 のテスト情報量制約を満たす 25 項目 ($L = 25$) のテスト構成を行った．また，OC の値は 1〜10 と 1 つづつ変化させて評価実験を行った．これらの条件は Fuchimoto et al. (2022)の評価実験と同様であり，実際に運用されている e-Testing の規模におけるテスト構成を模倣している．

結果を表 4 に示す．はじめに，提案手法は最大露出率および露出数の標準偏差を従来手法よりも抑えることができた．また，一部の条件において，従来手法(MCALIE)が露出数の標準偏差を抑えたが，他の手法に比べてテスト構成数が小さく，受検者数が 10 万人を超えるような試験では実用的でない．提案手法ごとに分析すると，提案手法(Det)は露出数の標準偏差を小さ

表 2. 実アイテムバンクの詳細.

アイテムバンクサイズ	識別力パラメータ a			難易度パラメータ b		
	範囲	平均	標準偏差	範囲	平均	標準偏差
87	0.15 ∼ 0.67	0.35	0.134	−2.09 ∼ 4.55	0.73	1.625
93	0.19 ∼ 0.69	0.43	0.122	−3.92 ∼ 3.61	−0.79	1.196
104	0.13 ∼ 1.10	0.59	0.213	−0.18 ∼ 4.55	1.50	1.188
141	0.24 ∼ 1.09	0.64	0.155	−1.41 ∼ 3.91	0.60	0.855
158	0.15 ∼ 3.08	0.44	0.255	−4.00 ∼ 4.00	−1.12	1.434
175	0.12 ∼ 0.93	0.39	0.139	−2.93 ∼ 3.12	−0.25	1.113
220	0.16 ∼ 0.92	0.46	0.155	−4.00 ∼ 2.82	−1.28	1.098
Total: 978	0.12 ∼ 3.08	0.46	0.198	−4.00 ∼ 4.55	−0.22	1.572

表 3. テスト情報量制約.

テスト情報量 $I(\theta)$ (下限値/上限値)				
$\theta = -2.0$	$\theta = -1.0$	$\theta = 0.0$	$\theta = 1.0$	$\theta = 2.0$
2.0/2.4	3.2/3.6	3.2/3.6	3.2/3.6	2.0/2.4

表 4. 従来手法と提案手法のテスト構成数および露出数.

アイテムバンクサイズ	OC	従来手法 (MCALIE) テスト構成数	最大露出率	露出数の標準偏差	従来手法 (HMCAPIP) テスト構成数	最大露出率	露出数の標準偏差	提案手法 (Det) テスト構成数	最大露出率	露出数の標準偏差	提案手法 (Stoch) テスト構成数	最大露出率	露出数の標準偏差	提案手法 (Det+Stoch) テスト構成数	最大露出率	露出数の標準偏差
2000 シミュレーション	1	1117	2.1	**4.3**	1169	2.1	4.5	1221	2.0	4.4	**1271**	2.0	4.6	1210	**1.9**	4.4
	2	7602	2.0	**27.6**	10354	2.0	37.7	10335	**1.7**	**34.5**	**10655**	1.9	37.7	10519	**1.7**	35.6
	3	27884	2.0	**104.6**	45313	2.0	168.0	38306	**1.6**	**125.1**	**52032**	1.7	184.5	43174	**1.6**	142.2
	4	62154	3.4	**276.7**	96166	3.0	401.1	97426	**1.6**	317.9	103962	**1.6**	374.6	**105152**	**1.6**	345.9
	5	90894	3.9	423.0	121260	3.5	537.7	120480	**1.6**	**393.1**	131656	**1.6**	476.0	**130896**	**1.6**	430.4
	6	94014	3.8	438.4	118804	3.6	533.8	**132223**	**1.6**	431.4	128816	**1.6**	465.8	117392	**1.6**	**386.1**
	7	93972	3.9	437.8	118007	3.7	530.2	120674	**1.6**	**393.7**	**131675**	**1.6**	476.0	122334	**1.6**	402.3
	8	93978	3.9	438.1	118967	3.7	534.5	123197	**1.6**	**402.0**	**129231**	**1.6**	469.2	122474	**1.6**	402.7
	9	93906	3.9	437.4	121893	3.6	545.9	124855	**1.6**	**407.4**	**128231**	**1.6**	463.5	119888	**1.6**	**394.3**
	10	94071	3.9	437.8	120728	3.7	539.7	123211	**1.6**	402.0	**132118**	**1.6**	477.7	121358	**1.6**	**399.1**
978 実データ	1	272	5.2	1.5	287	4.9	1.3	246	4.5	1.3	295	**4.4**	**1.2**	**301**	4.7	**1.2**
	2	1456	5.2	5.9	1415	5.2	5.8	1510	5.2	4.9	1511	4.7	5.0	**1574**	**4.6**	**4.6**
	3	7577	4.4	31.1	8856	5.4	34.7	8893	5.6	18.2	**9278**	3.9	25.7	8816	**3.8**	**16.7**
	4	27897	4.8	133.4	**32152**	4.8	153.8	26323	6.2	43.5	31254	3.3	81.8	24922	**3.0**	**35.6**
	5	51244	9.6	364.0	68395	8.7	452.0	65864	6.5	**109.8**	**73226**	3.1	198.1	62712	**3.0**	**86.4**
	6	85273	14.8	775.6	94341	13.9	809.4	100802	8.2	201.0	**102101**	3.1	280.3	93857	**3.0**	**124.2**
	7	93889	15.6	889.8	105266	14.8	945.4	100266	8.2	202.0	**108423**	**3.1**	298.6	99555	**3.1**	**128.2**
	8	94439	15.7	900.3	106071	14.8	952.9	102036	8.3	208.0	**106080**	**3.1**	295.4	99900	**3.1**	**128.1**
	9	94477	15.5	899.6	**106677**	14.7	968.7	105865	8.3	217.3	106538	**3.1**	293.3	100024	**3.1**	**128.4**
	10	94472	15.8	902.8	106195	14.7	950.9	105448	8.3	216.3	**111786**	**3.1**	307.7	100981	**3.1**	**129.8**

くできていることがわかる．これは，提案手法(Det)がロジスティック項目露出関数を用いた決定論的な重みを与えることで，露出数の低い項目ほど選ばれやすく，露出数の高い項目ほど選ばれにくくなるためである．一方で，提案手法(Stoch)は最大露出率を抑制できている．この理由は，提案手法(Stoch)が露出数の極端に高い項目に Big-M 法に基づいた重みを与えるので，その時点で露出数の高い項目を選択しないことにある．さらに，提案手法(Det+Stoch)は提案手法(Det)と提案手法(Stoch)の長所を持ち，最大露出率および露出数の標準偏差が最も抑えられる傾向にあった．

また，驚くべきことに，露出数を制御しているにも関わらず提案手法は従来手法(HMCAPIP)と同等以上のテストを構成できた．この理由は，露出数を制御しながらテスト構成する提案手法の方が従来手法よりも局所解(極大クリーク)に陥りにくく，多くのテストを構成できたと推測される．特に，従来手法では露出率の高い項目が存在することで，重複項目数の条件を満たす項目の組合せがテスト数を増加させるほど少なくなってしまい，局所解に陥るのかもしれない．実際に，最大露出率を抑えるためのペナルティだけを持つ提案手法(Stoch)が最もテスト構成数を増加させる傾向があった．そのため，提案手法(Stoch)が決定論的ペナルティを持つ 2 つの提案手法と比較して，最大露出率を抑えつつランダム性を持っており，テスト構成数を増加させていると推測できる．

次に，各手法の露出数の分布を分析するために，図 6 は(アイテムバンクサイズ，重複項目数条件)=(978,10)の条件における各手法の露出数のヒストグラムを表したものである．図 6(a)より，従来手法(HMCAPIP)は露出数の偏りが大きく，4500 回以上出題される項目が多数存在する．一方で，図 6(d)より，提案手法(Det+Stoch)では露出数の偏りがほとんどない．ただし，図 6(b)より，提案手法(Det)は 4500 回以上出題された項目が存在し，図 6(c)より，提案手法(Stoch)はほとんど出題されない項目が存在する．そのため，いずれかのペナルティだけでは，露出数の偏り改善に限界があった．

提案手法(Det)と提案手法(Stoch)で一部の項目に偏りが生じた原因は各項目のパラメータの特性値によるものと考えられる．特に，識別力パラメータの値が高い項目は情報量が高く，実行可能解に含まれる割合が大きいと推測される．そこで，(アイテムバンクサイズ，重複項目

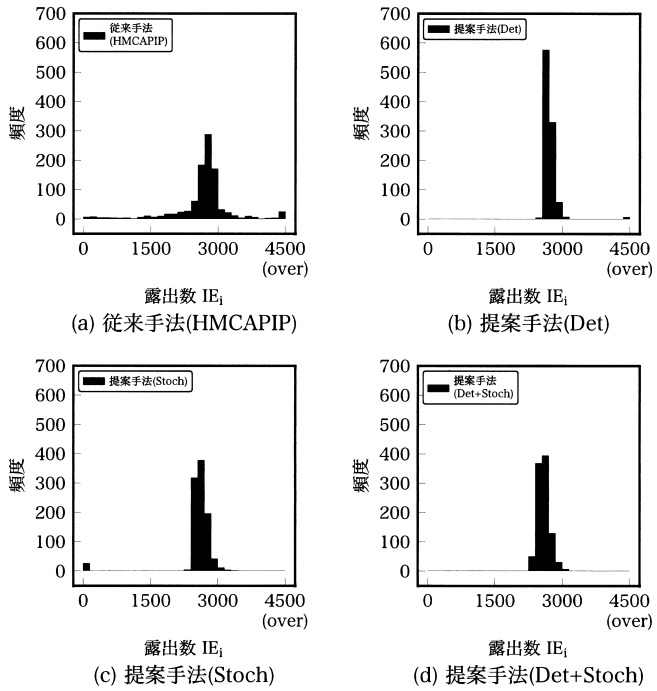

図 6. 露出数のヒストグラム(実データアイテムバンク，OC=10)．

数条件)=(978,10)の条件について，各手法における各項目の識別力パラメータとその項目の露出数を図 7 にプロットした．

図 7(a)より，従来手法は識別力パラメータの値が大きいほど，露出数の偏りが大きい．これは識別力パラメータの値が大きいほど情報量が高く，実行可能解に含まれる割合が大きいと考えられる．露出の高い項目は受検対策される恐れがあり，信頼性の低下につながる（Wainer, 2000)．特に，識別力の高い項目は作問が難しく貴重なため，過度な出題を避ける必要がある．

次に，提案手法ごとに分析すると，図 7(b)より，提案手法(Det)は 8000 回以上出題された識別力の高い項目が一つだけ存在する．この項目は過度な出題を避ける必要がある．また，図 7(c)より，提案手法(Stoch)はほとんど出題されない項目が複数存在する．これらの項目には識別力の高いものも含まれており，可能な限り活用することが望ましい．一方で，図 7(d)より，提案手法(Det+Stoch)は一様分布に近く理想的な露出数の分布にできた．

最後に，表 4 より，提案手法(Det)はシミュレーションアイテムバンクに限り，提案手法(Stoch)や提案手法(Det+Stoch)と同等に最大露出率を抑えられた．そこで，二つのアイテムバンクにおけるデータ特性との関連を分析するために，図 8 へ識別力パラメータのヒストグラムをプロットした．シミュレーションアイテムバンクは一様分布からデータを発生させたため，識別力の高い項目から低い項目まで均一に存在する．一方で，実データアイテムバンクは識別力の高い項目が少ない．したがって，提案手法(Det)は実データのように識別力パラメータの高い項目が不足している場合に最大露出率を抑制できないと考えられる．より各手法の特徴を分析するために，今後は項目パラメータの分布や他のテスト条件を変えた場合に，各手法が露出数の分布に与える影響を調査する．

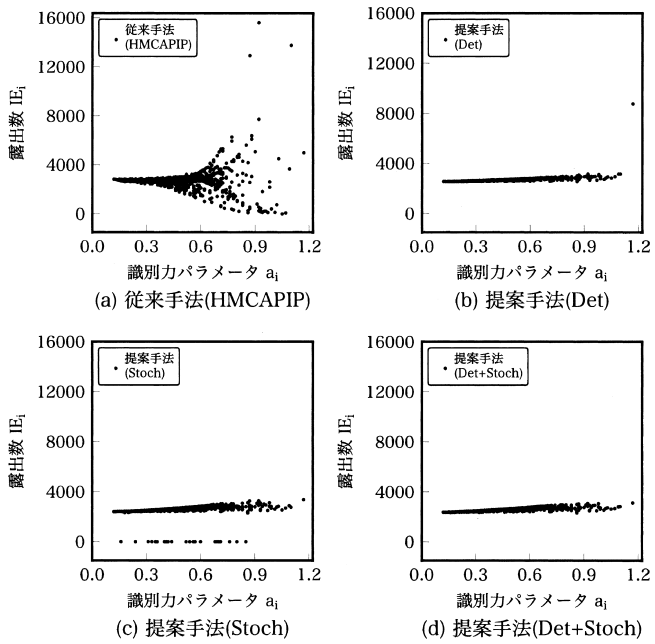

図 7. 識別力パラメータ a_i と露出数の散布図(実データアイテムバンク，OC=10)．

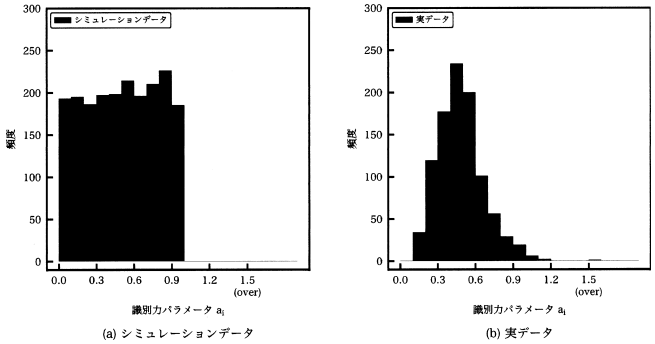

図 8. 識別力パラメータ a_i のヒストグラム．

6. むすび

本研究では，HCMAPIP 法における整数計画法の目的関数に露出数を所与としたロジスティック関数による 2 つのペナルティ項を追加した．提案手法ではこのロジスティック関数による決定論的・確率論的ペナルティ項を提案した．シミュレーション及び実データを用いた実験により，提案手法はテスト数を減らすことなく露出数の偏りを減らすことができた．これにより，e-Testing の信頼性向上が期待できる．

近年，測定精度を減少させずに問題数を減少できる技術として適応型テストが知られている(例えば，van der Linden, 2017; Kishida et al., 2023)．ここで，適応型テストとは受検者の能力を逐次的に推定しながら，その能力に応じて測定精度が最も高い項目を出題するテスト形式の一つである．しかし，同一能力の受検者には同一の項目群が出題される傾向にあり，特定の項

目群が頻繁に露出する傾向がある．また，各受検者の等質性が保証されていない．

この問題を解決する手法として，最大クリーク法による自動並行テスト構成技術（Fuchimoto et al., 2022）を用いた，二段階等質適応型テストが提案されている（Kishida et al., 2023）．この手法は従来の適応型と同等の測定精度を持ったまま問題数を減少でき，等質であるが露出数の分布が一様となるように出題できる．本研究の提案手法を二段階等質適応型テストに用いることで，露出数をより一様にできる可能性がある．

最後に，欧米では日本の共通テストに匹敵する Scholastic Aptitude Test（SAT）が全面的に e-Testing に移行する．また，我が国では 2024 年に電気通信大学において e-Testing による入試が開始される予定である（植野, 2023）．しかし，入学試験などで用いられる思考力を問う項目は所要時間が長く，十分な項目数を出題できずに測定精度が著しく低下する問題がある．この問題を解決するために，今後は各項目の所要時間を考慮し，制限時間内で等質であるが露出数の分布が一様となる自動並行テスト構成手法や二段階等質適応型テストへの拡張を検討する．これにより，e-Testing が制限時間内に各受検者の思考力を評価できることを目指す．

謝　辞

本研究の一部は科学研究費補助金（基盤研究（S），代表：植野真臣）「信頼性向上を持続する e テスティング・プラットフォームの開発」（19H05663）の助成を受けた．

参 考 文 献

Armstrong, R. D., Jones, D. H. and Wang, Z. (1994). Automated parallel test construction using classical test theory, *Journal of Educational Statistics*, **19**(1), 73–90.

Armstrong, R. D., Jones, D. H. and Kunce, C. S. (1998). IRT test assembly using network-flow programming, *Applied Psychological Measurement*, **22**(3), 237–247.

Boekkooi-Timminga, E. (1990). The construction of parallel tests from IRT-based item banks, *Journal of Educational Statistics*, **15**(2), 129–145.

独立行政法人情報処理推進機構 (2023). 【IT パスポート試験】情報処理推進機構, https://www3.jitec.ipa.go.jp/JitesCbt/index.html（最終アクセス日 2023 年 12 月 27 日）.

Fuchimoto, K., Ishii, T. and Ueno, M. (2022). Hybrid maximum clique algorithm using parallel integer programming for uniform test assembly, *IEEE Transactions on Learning Technologies*, **15**(2), 252–264.

Hetter, R. D. and Sympson, J. B. (1997). Item exposure control in CAT-ASVAB, *Computerized Adaptive Testing*: *From Inquiry to Operation* (eds. W. A. Sands, B. K. Waters and J. R. McBride), 141–144, American Psychological Association, Washington, D.C.

Ishii, T. and Ueno, M. (2015). Clique algorithm to minimize item exposure for uniform test forms assembly, *International Conference on Artificial Intelligence in Education*, 638–641, Springer, Switzerland.

Ishii, T. and Ueno, M. (2017). Algorithm for uniform test assembly using a maximum clique problem and integer programming, *International Conference on Artificial Intelligence in Education*, 102–112, Springer, Switzerland.

Ishii, T., Songmuang, P. and Ueno, M. (2013). Maximum clique algorithm for uniform test forms, *The 16th International Conference on Artificial Intelligence in Education*, 451–462.

Ishii, T., Songmuang, P. and Ueno, M. (2014). Maximum clique algorithm and its approximation for uniform test form assembly, *IEEE Transactions on Learning Technologies*, **7**(1), 83–95.

Kishida, W., Fuchimoto, K., Miyazawa, Y. and Ueno, M. (2023). Item difficulty constrained uniform adaptive testing, *International Conference on Artificial Intelligence in Education*, 568–573, Springer, Switzerland.

公益社団法人医療系大学間共用試験実施評価機構 (2023). 2023 年(令和 5 年)度 第 21 版 共用試験ガイドブック, https://www.cato.or.jp/e-book/21/index.html (最終アクセス日 2023 年 12 月 27 日).

Li, C.-M., Jiang, H. and Manyà, F. (2017). On minimization of the number of branches in branch-and-bound algorithms for the maximum clique problem, *Computers & Operations Research*, **84**, 1–15.

Lord, F. and Novick, M. (1968). *Statistical Theories of Mental Test Scores*, Addison-Wesley, Boston.

Nakanishi, H. and Tomita, E. (2008). An $O(2^{0.19171n})$-time and polynomial-space algorithm for finding a maximum clique, *IPSJ SIG Technical Report*, **2008**(6), 15–22.

Recruit (2023). Synthetic Personality Inventory (SPI), https://www.spi.recruit.co.jp/ (最終アクセス日 2023 年 12 月 27 日).

Samejima, F. (1977). Weakly parallel tests in latent trait theory with some criticisms of classical test theory, *Psychometrika*, **42**(2), 193–198.

Songmuang, P. and Ueno, M. (2011). Bees algorithm for construction of multiple test forms in e-Testing, *IEEE Transactions on Learning Technologies*, **4**, 209–221.

Stocking, M. L. and Lewis, C. (1998). Controlling item exposure conditional on ability in computerized adaptive testing, *Journal of Educational and Behavioral Statistics*, **23**(1), 57–75.

Stocking, M. L. and Lewis, C. (2000). Methods of controlling the exposure of items in CAT, *Computerized Adaptive Testing*: *Theory and Practice*, 163–182, Springer, New York City.

Sun, K.-T., Chen, Y.-J., Tsai, S.-Y. and Cheng, C.-F. (2008). Creating IRT-based parallel test forms using the genetic algorithm method, *Applied Measurement in Education*, **2**(21), 141–161.

Tomita, E., Yoshida, K., Hatta, T., Nagao, A., Ito, H. and Wakatsuki, M. (2016). A much faster branch-and-bound algorithm for finding a maximum clique, *International Workshop on Frontiers in Algorithmics*, 215–226, Springer, Cham.

Ueno, M. (2021). Ai based e-testing as a common yardstick for measuring human abilities, *2021 18th International Joint Conference on Computer Science and Software Engineering* (*JCSSE*), 1–5.

植野真臣 (2023). CBT の最前線, 情報処理, **64**(5), e1–e6.

Ueno, M., Fuchimoto, K. and Tsutsumi, E. (2021). E-testing from artificial intelligence approach, *Behaviormetrika*, **48**(2), 409–424.

van der Linden, W. J. (2005). *Linear Models for Optimal Test Design*, Springer, New York City.

van der Linden, W. J. (2017). *Handbook of Item Response Theory*: *Volume 3*: *Applications*, CRC Press, Florida.

van der Linden, W. J. and Adema, J. J. (1998). Simultaneous assembly of multiple test forms, *Journal of Educational Measurement*, **35**(3), 185–198.

van der Linden, W. J. and Choi, S. W. (2020). Improving item-exposure control in adaptive testing, *Journal of Educational Measurement*, **57**(3), 405–422.

van der Linden, W. J. and Reese, L. M. (1998). A model for optimal constrained adaptive testing, *Applied Psychological Measurement*, **22**(3), 259–270.

van der Linden, W. J. and Veldkamp, B. P. (2004). Constraining item exposure in computerized adaptive testing with shadow tests, *Journal of Educational and Behavioral Statistics*, **29**(3), 273–291.

Wainer, H. (2000). *CATS*: *Whither and Whence*, Educational Testing Service, New Jersey.

Williams, H. P. (1990). *Model Building in Mathematical Programming*, Wiley, New York City.

Automated Parallel Test Assembly Using Integer Programming with Item Exposure Penalties

Kazuma Fuchimoto and Maomi Ueno

Graduate School of Informatics and Engineering, The University of Electro-Communications

One feature of e-Testing is the automated test assembly of parallel test forms, by which each form has equivalent measurement accuracy, but with a different set of items. Unfortunately, the automated test assembly often causes a bias of item exposure. This difficulty of bias decreases the item and test reliability. To resolve this difficulty, this study examines a formulated test assembly problem as the objective function of integer programming with two logistic item exposure penalties: a deterministic penalty of logistic item exposure; and a stochastic penalty with logistic item exposure based on the Big-M method, which is a standard technique in mathematical programming. Numerical experiments demonstrate that the proposed methods reduce the bias of item exposure without decreasing the number of tests.

Key words: Automated test assembly, item response theory, e-Testing, item exposure problem, integer programming.