

# 項目露出を制約する 並行テスト構成アルゴリズムの提案と評価\*

2023年3月1日

## 1 まえがき

e テスティングとは、異なる問題から構成され、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。e テスティングを用いることで、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、受検者の同一精度での複数回受検が可能となる [2, 3, 4]。このようなテストを実現するために、e テスティングではテストに含まれる項目は異なるが、テスト間で性質が等質なテスト群を用いる。

このようなテストの概念として、同じ真の得点を測定し得点の測定誤差の分散も等しいテストを並行テストと Lord and Novick は定義した [5]。しかしこのようなテストは古典的テスト理論における概念であり、実現は困難であるため、項目反応理論を用いて並行テストを拡張した弱並行テストを Samejima は提案した [6]。能力値の測定誤差がテスト情報量の逆数に漸近的に一致することから弱並行テストの定義ではテスト情報量が等価なテストを並行テストとみなす。これにより異なる項目から構成された弱並行テストを同一能力の受検者が受検した場合、同一得点になることが保証される。e テスティングの普及に伴い、弱並行テストの定義に基づいたテスト自動構成手法が数多く提案されている [7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]。

一般に e テスティングでは、アイテムバンクと呼ばれる出題項目を管理するデータベースが利用される。アイテムバンクには出題項目の統計データが格納されており、所望のテ

---

\* 本論文は、電子情報通信学会論文誌 D vol.105 no.8 に掲載された同一著者による“項目露出を考慮した整数計画法による等質テスト構成” [1] に基づく Copyright ©2022 IEICE。本論文の資料は、電子情報通信学会論文誌 D vol.105 no.8 [1] で一部発表されたものであり、本論文の図の一部は電子情報通信学会の許諾のもと、[1] から再利用したものである。

ストの性質を満たす組合せを計算機で自動構成する．図 1 はテストの自動構成の概念図である．

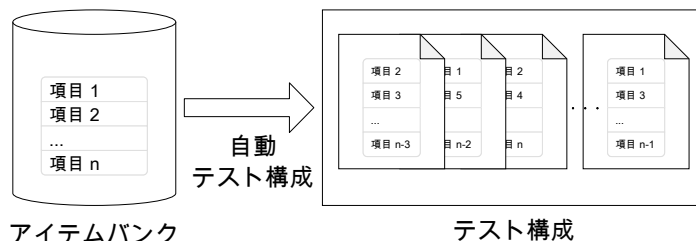


図 1 テストの自動構成

テストの自動構成は数理最適化問題として解かれる．例えば，Songmuang and Ueno (2010) は最適化問題の解探索手法の一つである Bees Algorithm を用いてテスト構成を提案した．この手法は我が国の国家試験である情報処理技術者試験で実際に使用されている [11] ．

Ishii, Songmuang, and Ueno (2014) はグラフ上で定義される最大クリーク問題に帰着してテスト構成を行う手法を提案した．具体的には与えられたアイテムバンクからテスト構成条件を満たすテストを生成し，グラフの頂点とみなす．二つのテスト（頂点）が等質条件を満たす場合に辺を引いたグラフ構造から最大クリーク探索することでテストを構成する [13, 20] ．この手法は理論的に最大のテスト構成を保証するが，アイテムバンクの項目数に対して構成可能な頂点（テスト）数が組合せ爆発的に増加するため，最大クリーク探索が困難である．この問題を解決するために，Ishii, Songmuang, and Ueno (2014) はグラフからランダムに部分グラフを選択し，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 (RndMCP 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している [14] ．最大クリーク探索はグラフの頂点集合を  $V$  とすると，最先端の最大クリーク探索手法 [21, 22] を用いても  $O(|V|^2)$  の空間計算量を必要とするため，最大で 10 万程度のテスト構成が限界であった．

RndMCP 法の空間計算量を緩和するために，Ishii and Ueno (2017) は第 1 段階目で RndMCP 法を用いてメモリの限界まで大きな最大クリークを探索した後，第 2 段階目で第 1 段階目で求めたクリークの全頂点と隣接する頂点を整数計画法により逐次的に探索することで空間計算量を  $O(|V|)$  に削減させる HybridRBP 法を提案した [15, 16] ．これにより 10 万を超えるテストを構成することが可能になった．しかし，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であった．そのため，Fuchimoto, Ishii, and Ueno (2022) は先行研究 [15, 16] で最も時間計算量

が大きい整数計画法を用いたクリーク探索を並列探索アルゴリズムに改善し、さらに多くのテスト構成を可能とした [17, 18] .

先行研究 [14, 16, 15, 17, 18] では構成されたテスト群は項目の露出数（出題回数）に偏りが生じる問題がある．露出数が多い項目は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 [23] . この露出数の偏りを軽減するために，Ishii and Ueno (2015) は RndMCP 法と整数計画法を用いてテスト構成し，その中から最も露出率（= 露出数の最大値 / テスト構成数）が小さいテスト構成を選択する手法を提案した [24] . 具体的には，探索した全てのテスト群（クリーク）を候補として保存しておき，最後にその候補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，従来手法よりも露出率を軽減することができた．しかし，Ishii and Ueno (2015) を用いても，特定の項目が過度に露出することが多い．

従来手法 [24] の第 1 段階目の RndMCP 法はランダム探索による最大クリーク探索で，探索範囲も限定的になり頂点集合における露出数に偏りが生じる問題がある．

この問題を解決するために，本論文では制約条件を満たす項目集合からなるべく一様に頂点を生成できるように整数計画問題の目的関数に一様分布に従う係数を掛け合わせて最大化する．これにより，第 1 段階目の頂点生成において項目露出の偏りを抑制する．

しかし，この整数計画法で制約を満たす解から一様に解を抽出するだけでは，項目露出のバイアス問題は完全には解決されない．そこでこの問題を解決するために，露出数上位  $s$  位までの項目をテスト構成ごとにアイテムバンクから除外し，テスト構成を行う手法を提案する．これにより，露出数上位の項目を除いたアイテムバンクからテスト構成するので，最大露出数が抑制される．テスト構成ごとにアイテムバンクから項目を除外していくと，いずれは構成条件を満たすようなテストを構成できなくなるが，その際に除外した項目をすべてアイテムバンクに戻す．

本提案手法は植野 (2021) [25]\*<sup>1</sup>を拡張した手法であり，植野 (2021) では整数計画法を用いて最大露出数の項目のみをアイテムバンクから除外しながらテストを構成していたが，本提案手法ではパラメータ  $s$  はテスト構成数と露出率に影響を与えるのでデータから決定する．また，植野 (2021) では係数が一様分布に従う目的関数を最大化することが露

---

\*<sup>1</sup> 植野 (2021) は同一著者の卒業論文であり，植野 (2021) では整数計画法を用いて露出数最大の項目をアイテムバンクから除外しながらテスト構成する手法を提案した．この提案では，一様分布に従う係数を掛け合わせた目的関数の最大化をする点と露出数最大の項目を除外する点が分割されておらず，露出率に与える影響をそれぞれで分析できていなかった．本論文では，これら 2 点を分割し比較実験によりそれぞれが与える影響を評価した．さらに，本論文では後者に関して露出最大の項目のみではなく，露出数上位  $s$  位までの項目をアイテムバンクから除外するようパラメータ化して提案し，実験により  $s$  の最適値を決定した．

出率に与える影響を分析できていない．本研究ではこの目的関数を用いること，この目的関数に加え露出数上位  $s$  位までの項目を除外することのそれぞれが露出率に与える影響を実験により評価した．さらに本研究では提案手法と RndMCP 法の出題項目の比較を通じ RndMCP 法で出題の偏りを引き起こす項目の分析を行った．

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最大露出数を抑制し露出率を改善したことが示された．また，いくつかの条件において，提案手法では従来手法よりも密なグラフ構造から最大クリーク探索を行うことができた．これにより，露出率だけでなく，テスト構成数も改善することができた．さらに，本研究では Choi, Lim, and Linden[26] の実アイテムバンクと実際のテストの制約を用い実験を行った．この実験により，実際のテストの仕様に基づく条件下でも提案手法では最大で 10 万を超える露出率が抑制されたテストを構成可能なことが示された．

## 2 テスト理論

### 2.1 並行テスト

同じ真の得点を測定し得点の測定誤差の分散も等しいテストを並行テストと Lord and Novick は定義した [5] . ただし , これは古典的テスト理論の概念であり , 現実には並行テストの実現は困難であるため , Samejima らは項目反応理論を用いて並行テストを拡張した弱並行テストを提案した [6] . 具体的には能力値の測定誤差が項目反応理論におけるテスト情報量の逆数に漸近的に一致するため , テスト情報量が等価なテストを弱並行テストと定義した . 弱並行テストの定義に基づいて数多くのテストの構成手法が提案されている [7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] .

### 2.2 項目反応理論

受検者得点の予測誤差は先行研究 ( 例えば [7, 11, 12, 27, 28, 29] ) において項目反応理論 ( Item Response Theory: IRT ) [30, 31] におけるテスト情報量の逆数で推定されている . IRT とは受検者の項目への正答確率をモデル化したものである . これにより , 異なる項目から構築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 .

IRT の中で最もよく使われる 2-パラメータロジスティックモデル ( 2-Parameter Logistic Model:2PLM ) では能力値  $\theta_j \in (-\infty, \infty)$  を持つ受検者  $j = (1, \dots, m)$  が項目  $i = (1, \dots, n)$  に正答する確率  $P_{2PLi}(\theta_j)$  を以下のようにモデル化する .

$$P_{2PLi}(\theta_j) = \frac{1}{1 + \exp[-1.7a_i(\theta_j - b_i)]} \quad (1)$$

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 , 困難度パラメータと呼ばれる項目パラメータである .

2PLM 以外でよく用いられる IRT モデルに , 3-パラメータロジスティックモデル ( 3-Parameter Logistic Model:3PLM ) がある . このモデルでは項目が選択問題であるときに偶然正答する可能性が考慮されている . 具体的には受検者  $j$  の項目  $i$  に対する正答確率は次式でモデル化される .

$$P_{3PLi}(\theta_j) = c_i + \frac{1 - c_i}{1 + \exp[-1.7a_i(\theta_j - b_i)]} \quad (2)$$

$c_i \in [0, 1]$  は項目  $i$  の当て推量パラメータと呼ばれ , この値が 0 のときは 2PLM と同一であるため , 3PLM モデルは 2PLM モデルを下位モデルとして含んでいる [2] .

他にも項目に対する反応が正誤の2値ではなく多値である場合に対応した，一般化部分採点モデル (Generalized Partial Credit Model:GPCM) が Muraki(1992) により提案されている [32] . 受検者  $j$  が項目  $i$  に対してカテゴリ ( 得点 )  $k (= 1, \dots, K)$  を得る確率は次式で表される .

$$P_{\text{GPC}ik}(\theta_j) = \frac{\exp \left[ \sum_{v=1}^k a_i(\theta_j - b_{iv}) \right]}{\sum_{l=1}^K \exp \left[ \sum_{v=1}^l a_i(\theta_j - b_{iv}) \right]} \quad (3)$$

$b_{ik}$  は項目  $i$  においてカテゴリ  $k-1$  からカテゴリ  $k$  を得る困難度を表し ,  $b_{i1} = 0$  である .

IRT では項目  $i$  において正答確率 ( 式 (1), (2),(3) ) を用いて計算したフィッシャー情報量を項目情報量  $I_i(\theta)$  と呼び , 2PLM, 3PLM, GPCM のそれぞれで以下のように定義する .

$$I_{2\text{PL}i}(\theta) = 1.7^2 a_i^2 P_{2\text{PL}i}(\theta)(1 - P_{2\text{PL}i}(\theta)) \quad (4)$$

$$I_{3\text{PL}i}(\theta) = \frac{1.7^2 a_i^2 (P_{3\text{PL}i}(\theta) - c_i)^2 (1 - P_{3\text{PL}i}(\theta))}{(1 - c_i)^2 P_{3\text{PL}i}(\theta)} \quad (5)$$

$$I_{\text{GPC}i}(\theta) = a_i^2 \left( \sum_{k=1}^K k^2 P_{\text{GPC}ik}(\theta) - \left( \sum_{k=1}^K k P_{\text{GPC}ik}(\theta) \right)^2 \right) \quad (6)$$

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 .

$$I(\theta) = \sum_{i \in T} I_i(\theta) \quad (7)$$

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ことが知られている [2] .

ただし , テストの自動構成手法 ( 例えば , [7, 11, 27, 28, 29, 33] ) ではテスト情報量における受検者の能力パラメータ  $\theta$  を  $\Theta = \{\theta_1, \dots, \theta_u, \dots, \theta_U\}$  のように幾つかの点でサンプリングし , 離散的に扱っている .

表 1 テスト情報量の上下限制約の例

$\theta = -2.0$	$\theta = -1.0$	$\theta = 0.0$	$\theta = 1.0$	$\theta = 2.0$
0.0/0.2	0.1/0.3	0.1/0.3	0.1/0.3	0.0/0.2

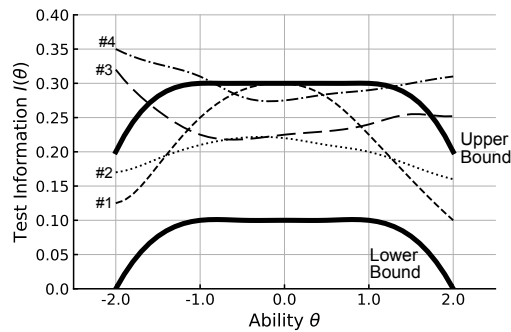


図 2 テスト情報量の上下限の例  
( Copyright ©2022 IEICE [1] 図 2 )

### 3 テストの自動構成アルゴリズム

#### 3.1 最大クリーク問題を用いた自動テスト構成

一般的にテストの自動構成では、以下の構成条件を満たすテストの集合として定義する。(例えば [8, 14, 20])

1. それぞれのテストでの受験者得点の予測誤差が等質である。
2. それぞれのテスト間の項目の重複数が一定値以下である。(以降、項目重複数条件と呼ぶ)

Ishii et al. (2014) はこの構成条件を満たす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手法 (ExMCP 法) を提案した [20]。ここでクリークはグラフの任意の 2 頂点が隣接する頂点集合である。

本手法では能力値パラメータ  $\Theta = \{\theta_1, \dots, \theta_u, \dots, \theta_U\}$  をサンプリングし、各点ごとにテスト情報量  $I(\theta_u)$  の上下限 ( $UB(\theta_u), LB(\theta_u)$ ) を設定し、全ての上下限制約を満たすテストを受検者得点の予測誤差が等質であるとする。

例えば図 2 は表 1 のテスト情報量の上下限制約を与えたときの概念図である。また、図

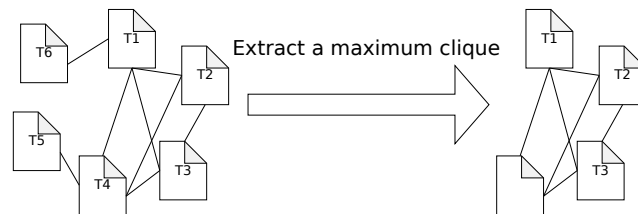


図3 最大クリーク問題を用いた自動テスト構成

中の#1~#4 は構成されたテストの情報量関数を描いたものである．図2のテスト#1 や#2 はテスト情報量の上下制限約を満たすため等質である．一方で，テスト#3 や#4 は上下制限約を満たさないため等質でない．

生成されるテスト候補を以下のグラフ構造とみなし，グラフ構造の中からクリークを探索する．

頂点： 項目重複数条件を除いたテスト構成条件を満たす，すべてのテストを頂点とする．

辺： 二つの頂点に対応するテストが項目重複数条件を満たす場合，二つの頂点間に辺を引く．

このグラフが持つクリーク中の任意の二頂点は隣接しているため，テスト構成条件を満たす．したがって，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，頂点数が最大のクリークが最大のテストとなる（図3）．

以上から，無向グラフ  $G = (V, E)$  の頂点の集合を  $V$ ，辺の集合を  $E$  としたとき，最大クリーク  $C$ （最大のテスト構成）は次のように定式化できる．

テスト構成のための最大クリーク問題

**variables**      $C \subseteq V$   
**maximize**      $|C|$   
**subject to**      $\forall v, \forall w \in C, \{v, w\} \in E$

\*ここで  $\{v, w\} \in E$  は頂点の組  $v, w$  が次の条件で引かれた辺を意味する  
 $(|v \cap w| \leq \text{重複項目数の上限値})$  .

この手法は理論的に最大数のテストの構成が可能であるが，最先端の最大クリーク探索手法 [21, 22] を用いても，時間計算量  $O(2^{|V|})$ ，空間計算量  $O(|V|^2)$  を必要とする．ここで，頂点数  $|V|$  はテスト構成条件を満たすテストの総数となるが，その数はアイテムバンクの項目数  $n$  と比較して組合せ爆発的に増加する．ゆえに，現在実施されているような数百~千以上のアイテムバンクから最大のテストの構成を厳密に行うことは困難である．



ExMCP 法の計算コストの問題を緩和するため, Ishii, Songmuang, and Ueno (2014) は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 [14] (以降 RndMCP 法と呼ぶ). ExMCP 法 [20] の問題点はテスト構成条件を満たす頂点が爆発的に増加し, グラフの探索空間が莫大となることである. そのため, RndMCP 法ではテスト構成候補グラフ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全体の最大クリークを近似的に探索する.

この手法は  $L_1$  個の頂点からなる部分グラフにおける最大クリーク探索を時間  $L_2$  だけ行うため, 時間・空間計算量をそれぞれ  $O(L_2), O(L_1^2)$  に緩和する. また, 最大クリーク探索は時間  $CT$  だけ繰り返す. これらのパラメータは計算機環境に合わせて任意に設定できるため, ExMCP 法 [20] の時間・空間計算量  $O(2^{|V|}), O(|V|^2)$  に対して格段に扱いやすい.

この手法により一般的な規模 (500 ~ 2000 項目程度) のアイテムバンクから最大で 10 万程度のテストを生成できた.

### 3.2 整数計画法を用いたテスト構成

RndMCP 法ではグラフの頂点数を  $|V|$  としたとき, 最先端の最大クリーク探索アルゴリズム [21, 22] を用いたとしても空間計算量を  $O(|V|^2)$  必要とするため 10 万個程度のテスト構成が上限であった. そこで, Ishii et al. (2017) は RndMCP 法と整数計画法を組み合わせた HybridRBP 法を提案した [15, 16].

HybridRBP 法では現在探索中のクリーク  $C$  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. 本手法では現在探索中のクリークに隣接する頂点のみを保存するため, グラフの頂点数を  $|V|$  とすると, 計算に必要な空間計算量は  $O(|V|)$  に軽減される. ただしこの探索は  $O(|V| \cdot 2^n)$  の時間計算量を必要とするため, RndMCP 法の最大クリーク探索の時間計算量  $O(L_2)$  に大幅に劣る. そこで RndMCP 法により計算機のメモリの限界の頂点数  $L_1$  を持つグラフで最大クリーク探索を行ってから, 整数計画法を用いる手法に切り替えることで探索効率を改善する [15, 16]. 具体的には次の整数計画問題でクリークに隣接する頂点を探索する.

最大クリーク探索のための整数計画問題

**variables**

$$x_i = \begin{cases} 1 & i \text{ 番目の項目がテストに含まれる} \\ 0 & \text{それ以外} \end{cases} \quad (8)$$

**maximize**

$$\sum_{i=1}^n \lambda_i x_i \quad (9)$$

**subject to**

$$\sum_{i=1}^n x_i = M (\text{テスト項目数}) \quad (10)$$

$$LB_{\theta_u} \leq \sum_{i=1}^n I_i(\theta_u) x_i \leq UB_{\theta_u} \quad (11)$$

$(u = 1, \dots, U)$

$$\sum_{i=1}^n X_{i,r} x_i \leq OC (\text{項目重複上限数}) \quad (12)$$

$(r = 1, \dots, |C|)$

$$X_{i,r} = \begin{cases} 1 & i \text{ 番目の項目が} \\ & C \text{ 中の } r \text{ 番目のテストに含まれる} \\ 0 & \text{それ以外} \end{cases}$$

制約条件の式 (12) はクリーク  $C$  の全頂点と隣接するための条件である。

目的関数に含まれる  $\lambda_i (i = 1, 2, \dots, n)$  は互いに独立な  $[0, 1)$  の連続一様分布であり、本問題が解かれるごとにリサンプリングされ毎回ランダムにテストが構成される。この定式化は Belov (2008) [9] で用いられたランダムにテスト構成を行う整数計画問題への定式化を項目重複数について一般化したものとなっている [15]。

この手法の空間計算量は内部で使用する整数計画法の空間計算量が無視できるとすると、 $O(|V|)$  となる。RndMCP 法と比較して空間計算量が  $O(|V|^2)$  から  $O(|V|)$  に減少しているため、構成可能なテストの上限は大きくなる。これにより、乱択法では 10 万個のテスト構成が上限であったが、HybridRBP 法では 10 万を超えるテスト構成が可能になった。

### 3.3 露出率を軽減するテスト構成手法

先行研究 [14, 16, 15] では、ある項目がテスト構成全体で出題される回数（露出数）に偏りが生じる問題がある。露出数が多い項目は受検者間で共有されやすく、その項目の信頼性が失われやすくなる [23]。ここで、テスト群（クリーク  $C$ ）に含まれる項目  $i$  の露出数  $IEC_i$  は以下のように表される。

$$IEC_i = \sum_{r=1}^{|C|} X_{i,r} \quad (13)$$

$$X_{i,r} = \begin{cases} 1 & (\text{項目 } i \text{ がテスト } r \text{ に含まれる}) \\ 0 & (\text{項目 } i \text{ がテスト } r \text{ に含まれない}) \end{cases}$$

また、テスト構成  $C$  における最大露出数を

$$IEC_{\max} = \max_{i=1 \dots n} IEC_i \quad (14)$$

で表す。このときテスト構成  $C$  における項目露出率を

$$IEC_{\text{rate}} = \frac{IEC_{\max}}{|C|} \quad (15)$$

とする。

Ishii and Ueno (2015) はこの露出率を軽減するテスト構成を提案した [24]。本手法は RndMCP 法 (3.2 節) と整数計画問題を用いた最大クリーク探索 (3.3 節) を用いてテスト構成し、そのテスト構成の中で露出率が最小となるようなテスト構成を選択する。

具体的にはアルゴリズム 1 により、テスト構成を行う。以降、この手法を Maximum Clique Algorithm with Low Item Exposure (MCALIE) 法と呼ぶ。

---

**Algorithm 1** MCALIE 法

---

**Require:** アイテムバンク , テスト構成条件

**Ensure:** テスト群

```
1: procedure MCALIE( $L_1, L_2, CT', \alpha, CT$ )
2:    $ST := current\ time$ 
3:   /* initialize */
4:   global  $C := RndMCP(L_1, L_2, CT')$ 
5:   global  $C_{candidate} := \emptyset$ 
6:   while ( $current\ time - ST < CT$ ) do
7:     /* add step */
8:      $Sol := IPSolve(itemBank, C)$                                 ▶ 式(9) ~ 式(12) を解く
9:     if  $Sol \neq \emptyset$  then                                    ▶ IP が解けた場合
10:       $C := C \cup \{Sol\}$ 
11:       $C_{candidate} := C_{candidate} \cup \{C\}$ 
12:     else                                                        ▶ IP が解けない場合
13:       DeleteStep( $|C|, \alpha$ )
14:     end if
15:   end while
16:   /* output */
17:    $C_{res} := C$ 
18:   for  $C_{cand}$  in  $C_{candidate}$  do
19:     if  $\frac{E_{C_{res}}}{|C_{res}|} \leq \frac{E_{C_{cand}}}{|C_{cand}|}$  then
20:        $C_{res} := C_{cand}$                                           ▶ 露出率が最小のテスト群を選択する
21:     end if
22:   end for
23:   return  $C_{res}$ 
24: end procedure
25: procedure DeleteStep( $cliqueSize, \alpha$ )
26:   /* delete step */
27:    $count := 0$ 
28:   while  $count < (cliqueSize \times \alpha)$  do
29:      $C := C \setminus \{c \in C\}$                                 ▶ 最大露出項目を持つテストから除外する
30:      $count++$ 
31:   end while
32:    $C_{candidate} := C_{candidate} \cup \{C\}$ 
33: end procedure
```

---

MCALIE 法は大きく分けて “initialize”, “add step”, “delete step”, “output” から構成される .

“initialize” では RndMCP 法によりメモリ限界  $L_1$  の制限下で最大クリーク  $C$  を探索

表2 MCALIE 法の再現実験の結果

Pool Size	OC	RndMCP			MCALIE		
		$ C $	$IEC_{max}$	$IEC_{rate}$	$ C $	$IEC_{max}$	$IEC_{rate}$
978 (actual)	0	18	1	5.6%	<b>36</b>	1	<b>2.8%</b>
	5	45790	5177	11.3%	<b>51644</b>	5342	<b>10.3%</b>
	10	100000	16310	16.3%	<b>102862</b>	16698	<b>16.2%</b>

する。

“add step”では整数計画法により最大クリーク  $C$  に隣接する頂点を逐次的に探索し  $C$  に追加することで、より大きなテストを生成する。生成したテスト  $C$  は  $C_{candidate}$  に保存する。

“delete step”では“add step”で整数計画法が解けなかった場合、 $C$  から一定割合 ( $\alpha$ ) だけ頂点を除外する。この際、最大露出の項目を含むテストから削除することで露出数を減らしている。削除によって変更されたクリーク  $C$  は  $C_{candidate}$  に保存する。

“output”では生成したテストの集合  $C_{candidate}$  から露出率  $\frac{Exc}{|C|}$  が最も低いテスト  $C$  を選択し出力する。

これらの提案により、従来手法 [7, 10, 11, 14] よりも露出率が軽減されている [24]。

### 3.4 MCALIE 法の問題

MCALIE 法 [24] は構成したテスト群を保存しておき、最後にその中から最も露出率が低いテスト群を選択することで露出率を軽減している。しかし、テストを構成する際に露出を考慮していないため、特定の項目が過度に露出（出題）される問題を解消できていない。

この従来手法の問題を明らかにするために、MCALIE 法の再現実験を行なった。ここで、再現実験は 978 項目からなる実際に運用されたアイテムバンクを用い、テスト項目数を 25、重複項目数  $OC=\{0, 5, 10\}$  として実験を行った。パラメータ値は RndMCP 法が  $L_1 = 100000, L_2 = 4hr, CT = 4hr$ 、MCALIE 法が  $L_1 = 100000, L_2 = 4hr, CT' = 4hr, CT = 24hr$  である。

結果を表 2 に示す。MCALIE 法は RndMCP よりもテスト構成数が増大し、露出率は減少はしたものの、依然として露出率は大きい。図 4 は  $OC = 5$  の再現実験で構成されたテストの項目露出数を昇順に並べ替えたものである。具体的には縦軸に露出数、横軸は露出数の低い順に並べ替えられた項目を示している。図 4 のように一部項目は露出数が非常に

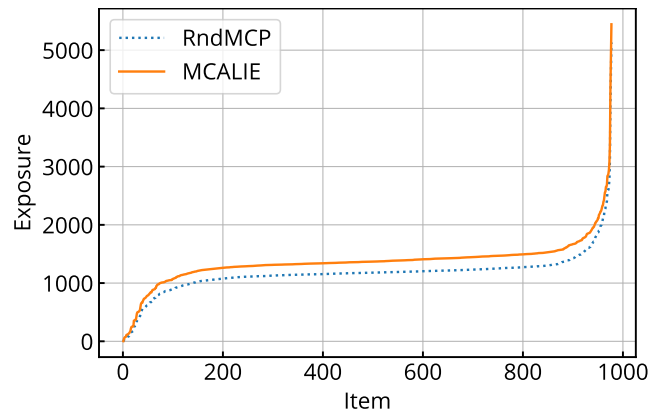


図4 MCALIE法の露出数の分布  
 (Copyright ©2022 IEICE [1] 図4)

大きく(グラフの右端), テスト全体の10%以上に出題されている。MCALIE法も特定の項目が優先的に選択され, 過度な露出を引き起こしていることがわかる。

## 4 提案手法

MCALIE 法 [24] の第 1 段階目の RndMCP 法はランダム探索による最大クリーク探索であり、頂点集合における露出数の偏りが生じる問題がある。この問題を解決するために、本論文では整数計画法を用いて頂点生成する。提案手法の整数計画法ではテストの構成条件を満たす項目集合のうち目的関数（式（9））が最大のものを選択する。この目的関数の係数  $\lambda_i$  は一様分布に従う乱数であるため、制約を満たす頂点集合から頂点がほぼ一様に選択される。このため、第 1 段階目の頂点生成において項目露出の偏りを改善することができる。

しかしこの整数計画法は項目ごとの露出数の一様性は考慮していないため、依然として特定の項目が過度に出題されてしまうバイアス問題がある。この問題を解決するために、頂点集合における露出数上位  $s$  位までの項目をテスト構成ごとにアイテムバンクから除外し、テスト構成を行う手法を提案する。これにより、露出数上位の項目を除いたアイテムバンクからテスト構成するので、最大露出数が抑制され、テスト構成の項目露出数が一様になることが期待される。

具体的には、第 1 段階目の頂点生成において項目重複数条件（式（12））を除いた整数計画問題（式（9）～式（11））を解き、頂点生成し頂点集合に追加する。頂点生成後に、すべての項目のうち頂点集合における露出数上位  $s$  位までの項目をアイテムバンクから除外する。以上を頂点集合の大きさが  $L_1$  になるまで逐次的に繰り返し頂点を生成する。ただし、新たに頂点を生成できない場合は今までに除外した項目をすべてアイテムバンクに戻す。その後、RndMCP 法と同様に頂点集合から構築されるグラフの最大クリークを探索する。

提案手法では露出数の大きい項目が除外されたアイテムバンクからテスト構成するため、図 4 のように特定の項目が選択され続けることを防ぎ、項目がテスト群に均等に追加され、従来手法よりも露出数の偏りを軽減できると期待される。

本研究では、グラフの頂点集合（テスト集合）を  $V$  とし、式（16）の項目  $i$  の  $V$  における露出数  $IEV_i$  を導入する。

$$IEV_i = \sum_{r=1}^{|V|} X_{i,r} \quad (16)$$

$$X_{i,r} = \begin{cases} 1 & (\text{項目 } i \text{ がテスト } r \text{ に含まれる}) \\ 0 & (\text{項目 } i \text{ がテスト } r \text{ に含まれない}) \end{cases}$$

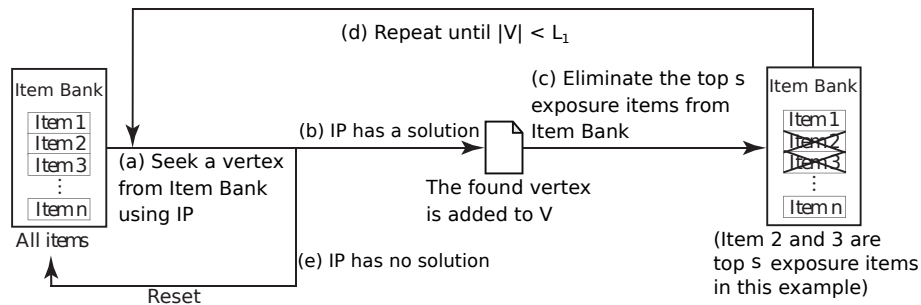


図5 第1段階目の提案手法  
( Copyright ©2022 IEICE [1] 図5 )

提案手法は概要図5のように， $IEV_i$  を用いて以下の手順に従い頂点を生成する．

- (a) 整数計画問題（式（9）～式（11））を解く．
- (b) 整数計画問題が解けた場合は，その解を  $V$  に追加する．
- (c) 露出数  $IEV$  が上位  $s$  位までの全ての項目をアイテムバンクから除外する．（図5の  
は項目2と3が該当する．）
- (d) (a)～(c) の手順を  $L_1$  個の頂点を生成するまで繰り返す．
- (e) 整数計画問題が解けない場合は，除外した項目を全てアイテムバンクに戻す．

整数計画問題は NP 困難であるため，一定の計算時間 IPT で解を得られない場合は問題が解けないとみなし，(e) で今までに除外したすべての項目をアイテムバンクに戻す．

頂点生成後は RndMCP と同様にして項目重複数条件を満たす2頂点間に辺を引いてグラフを構築し，そのグラフにおいて，最大クリーク探索を行う．

この提案手法を組み込んだ RndMCP 法のアルゴリズムを Algorithm2 に示す．

---

#### Algorithm 2 選択可能な項目を制限した RndMCP 法

---

**Require:** アイテムバンク，テスト構成条件

**Ensure:** テスト群

- 1: **procedure** LimitItemRndMCP( $L_1, L_2, CT, s$ )
- 2:      $C := \emptyset, C_{\max} := \emptyset$
- 3:      $ST := \text{current time}$
- 4:     **while** ( $\text{current time} - ST < CT$ ) **do**
- 5:         /\* Step1 \*/
- 6:          $V := \emptyset$



```

7:      n := itemBank.size
8:      for i ← 0 to n - 1 do
9:          IEV[i] := 0
10:     end for
11:     allowedItems := アイテムバンクの全項目
12:     while |V| < L1 do
13:         Sol := IPSolve(allowedItems)
14:         ▶ allowedItems のみからなるアイテムバンクに対し式 (9) ~ 式 (11) の整数計画問
           題を解く
15:         if Sol = ∅ then                                     ▶ 整数計画問題が解けない場合
16:             allowedItems := アイテムバンクの全項目
17:             break
18:         end if
19:         V := V ∪ {Sol}
20:         for each i ∈ Sol do
21:             IEV[i] ++
22:         end for
23:         allowedItems := IEV の上位 s 位以外の項目
24:     end while
25:     /* Step2 */
26:     G = (V, E) グラフ構築                                     ▶ 二頂点が重複項目数条件を満たす場合, 辺を引く
27:     /* Step3 */
28:     C := MCP(G, L2)                                         ▶ G の最大クリークを時間 L2 だけ探索
29:     if |Cmax| < |C| then
30:         Cmax := C
31:     end if
32: end while
33: return Cmax
34: end procedure

```

step1 では最大クリーク探索するための部分グラフの頂点を  $L_1$  個生成する。その頂点集合を  $V$  とする。allowedItems は IEV が上位  $s$  位までの項目をアイテムバンクから除いた項目集合である。項目重複数条件 (式 (12)) を除いた整数計画問題 (式 (9) ~ 式 (11)) を解き、頂点 (Sol) を生成する。整数計画問題の解を求めることができない場合は、allowedItems に除外した項目を戻し、再度整数計画問題を解く。頂点に含まれる項目について IEV をインクリメントする。

step2 では step1 で生成した頂点集合  $V$  からグラフ  $G$  を構築する。具体的には項目重複数条件を満たす 2 頂点間に辺を引きグラフを構築する。

表3 アイテムバンクのパラメータ分布

Pool Size	parameter $a$		parameter $b$	
	Mean	SD	Mean	SD
500	0.49	0.28	0.05	1.06
1000	0.49	0.28	0.01	1.00
2000	0.51	0.29	0.04	1.00
978(actual)	0.43	0.20	-0.22	1.16

step3 では step2 で構築したグラフ  $G$  において最大クリーク探索をして、テスト構成をする。

## 5 評価実験

提案手法の有効性を示すために、評価実験を行う。最初に、RndMCP 法と提案手法の整数計画法を比較する。次に、提案手法の露出数上位  $s$  位までの項目を除外するアルゴリズムのパラメータの最適化を行う。最後に、第 2 段階目で MCALIE 法と HybridRBP の第 1 段階目の手法として RndMCP 法と提案手法を用いたものとで比較し、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 3.50GHz, RAM 128GB) である。

本論文の実験では、先行研究 [24] の実験と同様に実際に運用されていた実アイテムバンク及びシミュレーションアイテムバンクを用いた。さらに、Choi et al.(2022)[26] の実験で用いられている実アイテムバンクと実際のテストの制約も課して実験を行った。

先行研究 [24] のアイテムバンクは実アイテムバンクが 978 項目、シミュレーションアイテムバンクは 500, 1000, 2000 項目を持ち、各項目は 2PLM である。項目の識別力パラメータ  $a$  を  $\log_2 a \sim N(0, 1^2)$ 、困難度パラメータ  $b$  を  $b \sim N(0, 1^2)$  として実アイテムバンクのパラメータ値の分布に近づけるよう発生させた。実アイテムバンク及び生成したシミュレーションアイテムバンクのパラメータ値の分布の詳細は表 3 の通りである。

テストの構成条件は前述したアイテムバンクから、表 4 のテスト情報量の上下制限約を満たす 25 項目のテスト構成とした。本条件は実際に運用された e テスティングにおけるテスト構成条件である。

Choi et al.(2022)[26] の実アイテムバンクは science と呼ばれるデータセットに含まれ、918 項目の 3PLM と 82 項目の GPCM の計 1000 項目からなる。3PLM の項目は識別力パ

表 4 実験で用いたテスト情報量の上下限制約

テスト項目数	$\theta = -2.0$	$\theta = -1.0$	$\theta = -0.0$	$\theta = 1.0$	$\theta = 2.0$
25	2.0/2.4	3.2/3.6	3.2/3.6	3.2/3.6	2.0/2.4
30	2.44/2.84	3.88/4.28	3.88/4.28	3.88/4.28	2.44/2.84

表 5 science データセットのアイテムバンクにおける 3PLM の項目のパラメータ分布

parameter $a$		parameter $b$		parameter $c$	
Mean	SD	Mean	SD	Mean	SD
0.83	0.34	-0.40	1.23	-0.15	0.09

表 6 science データセットのアイテムバンクにおける GPCM の項目のパラメータ分布

parameter $a$		parameter $b_{i2}$		parameter $b_{i3}$	
Mean	SD	Mean	SD	Mean	SD
1.22	0.39	0.34	1.52	0.25	1.86

ラメータ  $a$  , 困難度パラメータ  $b$  , 当て推量パラメータ  $c$  を持ち , それぞれのパラメータの詳細を表 5 に示す . GPCM の項目は識別力パラメータ  $a$  とカテゴリ ( 得点 ) 1 から 2 , 2 から 3 を得る困難度パラメータ  $b_{i2}, b_{i3}$  を持ち , その詳細は表 6 の通りである .

制約条件は実際のテストの仕様を基に設定された表 12 の 36 制約である . 表 12 の CONSTRAINT\_ID 列は制約の ID , TYPE 列は制約の種類 , CONDITION 列は制約の条件式 , LB ( UB ) 列は CONDITION 列を満たすべき項目数の下限 ( 上限 ) 値 , COUNT 列は制約を満たす項目数を示している . ただし Choi et al.(2022)[26] では整数計画問題が実行不可能に陥らないようすべての制約は課さずにテスト構成することを推奨している . 具体的には表 12 の CONSTRAINT\_ID が C1~C13, C21~C31 の制約のみを用いてテスト構成を行う . C1 はテスト項目数制約 ( 式 (10)) であり , 30 項目のテストを構成する条件である . C2~C13, C21~C31 の制約はカテゴリカル制約と呼ばれる . 実際のテストでは項目を難易度・教育目的・種類などの属性とその値でカテゴリに分類することがある . カテゴリ内で所望の数の項目を出題する制約がカテゴリカル制約であり , 以下の式で表される .

$$LB_{cat} \leq \sum_{i \in V_{cat}} x_i \leq UB_{cat} \quad (\text{カテゴリカル制約}) \quad (17)$$

$$(cat = C_1, \dots, C_g)$$

$cat$  はカテゴリを表し,  $V_{cat}$  はカテゴリ  $cat$  に属する項目の集合である.  $V_{cat}$  に属する項目が  $LB_{cat}$  項目以上  $UB_{cat}$  項目以下のテストを構成する. 表 12 では項目の属性は LEVEL, STANDARD, OBJECTIVE, DOK, TYPE があり, その属性の値に応じて CONDITION 列の条件でカテゴリに分類する. このカテゴリに属する項目数が LB 列以上 UB 列以下のテストを構成する. 本実験でも Choi et al.(2022)[26] にならい項目数制約 C1 とカテゴリカル制約 C2~C13, C21~C31 を課し, テスト情報量の上下制限は表 4 のテスト項目数 30 の条件でテスト構成を行った.

いずれのアイテムバンクに対しても, 項目重複数条件は先行研究 [24] と同様に  $OC = \{0, 5, 10\}$  の 3 通りの条件によって評価する. なお, 従来手法 [24] 及び提案手法の整数計画法の求解には CPLEX12.9[34] を用い, 整数計画緩和の解との相対ギャップが  $10^{-4}$  以下で打ち切るようにした (デフォルトのオプション). さらに整数計画問題の計算時間の上限を  $IPT = 60\text{sec}$  と設定した.

## 5.1 RndMCP 法と整数計画法の比較実験

第 1 段階目の頂点生成において, 提案手法の整数計画法を用いることで露出数の偏りが改善されることを示すために, RndMCP 法 [14] と比較実験を行った.

パラメータ値は  $L_1 = 100000$ ,  $L_2 = 4hr$ ,  $CT = 4hr$  とした. ただし,  $n = 978, 1000$  の実データの場合は計算時間  $CT = 4hr$  以内に頂点を  $L_1$  個生成できなかったため,  $CT = 8hr$  と計算時間を延長した.

ここで, 各項目の露出数  $IEC_i$  の平均を

$$IEC_{\mu} = \frac{\sum_{i=1}^n IEC_i}{n} \quad (18)$$

とし, 各項目の露出数  $IEC_i$  の標準偏差を

$$IEC_{\sigma} = \sqrt{\frac{\sum_{i=1}^n (IEC_i - IEC_{\mu})^2}{n}} \quad (19)$$

とする.

重複項目数を  $OC$ , テスト構成数を  $|C|$ , 最大露出数を  $IEC_{\max}$ , 露出率を  $IEC_{\text{rate}}$  とし, 露出数の標準偏差  $IEC_{\sigma}$  とともに結果を表 7 に示す. なお, 露出数の平均値  $IEC_{\mu}$  については手法間でテスト構成数が異なると, 単純に比較することができないため表には載せていない.

シミュレーションアイテムバンクと実アイテムバンク  $n = 978$  について,  $OC = 0$  以外の条件下で整数計画法のほうが露出数の標準偏差及び露出率が小さくなった. 提案手法で

表 7 RndMCP 法と整数計画法の比較実験の結果

Pool Size	OC	RndMCP				提案手法 ( 整数計画法 )			
		$ C $	$IEC_{\sigma}$	$IEC_{max}$	$IEC_{rate}$	$ C $	$IEC_{\sigma}$	$IEC_{max}$	$IEC_{rate}$
500	0	<b>10</b>	0.5	1	<b>10.0%</b>	<b>10</b>	0.5	1	<b>10.0%</b>
	5	4371	53.1	378	8.6%	<b>4952</b>	50.5	402	<b>8.1%</b>
	10	<b>99981</b>	1729.2	13006	13.0%	99979	1153.2	7259	<b>7.3%</b>
1000	0	17	0.5	1	5.9%	<b>18</b>	0.5	1	<b>5.6%</b>
	5	46190	430.5	3374	7.3%	<b>52023</b>	389.1	2206	<b>4.2%</b>
	10	<b>100000</b>	990.7	8767	8.8%	<b>100000</b>	755.4	4139	<b>4.1%</b>
2000	0	<b>32</b>	0.5	1	<b>3.1%</b>	31	0.5	1	3.2%
	5	<b>96773</b>	456.1	3833	4.0%	88651	325.2	1674	<b>1.9%</b>
	10	<b>100000</b>	472.6	4044	4.0%	90917	333.7	1707	<b>1.9%</b>
978 (actual)	0	18	0.5	1	5.6%	<b>19</b>	0.5	1	<b>5.3%</b>
	5	45790	368.2	5177	11.3%	<b>54964</b>	251.7	2303	<b>4.2%</b>
	10	<b>100000</b>	985.3	16310	16.3%	<b>100000</b>	488.2	4301	<b>4.3%</b>
1000 (actual)	0	0	-	0	-	10	0.5	1	<b>10%</b>
	5	0	-	0	-	<b>7992</b>	111.9	944	<b>11.8%</b>
	10	1	0.2	1	100.0%	<b>99989</b>	1526.7	12685	<b>12.7%</b>

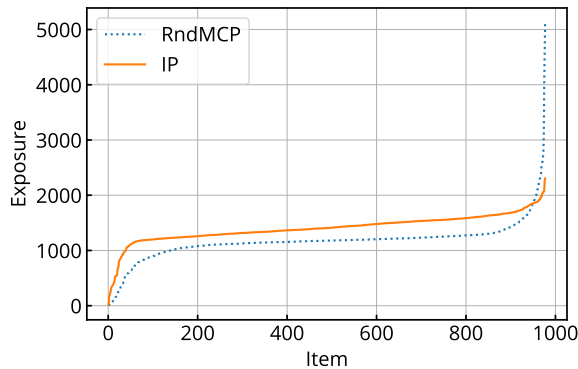


図 6 RndMCP 法と整数計画法の露出数の分布  
( Copyright ©2022 IEICE [1] 図 6 )

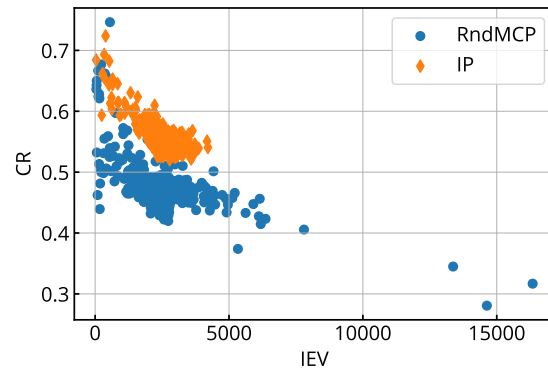


図 7 露出数とクリークの関係  
( Copyright ©2022 IEICE [1] 図 7 )

は制約を満たす頂点集合から頂点をなるべく一様に生成するように係数が一様分布に従う目的関数を最大化することで、露出率が改善されることが示された。

図 6 は  $n = 978, OC = 5$  における RndMCP 法と提案手法 ( 整数計画法 ( IP )) によるテスト構成の露出数の分布である。縦軸は露出数、横軸は項目を表し、項目は露出数の昇順

に並び替えている．図 6 から整数計画法により露出数の偏りが改善されたことが確認できる．しかし，特定の項目に偏って露出される問題は依然として存在する．図 6 の整数計画法では特定の項目（グラフの右端）が 2303 個のテストに含まれ，テスト全体の 4.2% に出題される．本提案手法の整数計画問題により制約を満たす頂点集合からなるべく一様になるように頂点を生成できるが，各項目ごとの露出数の一様性については考慮していないので，依然として特定の項目に偏って露出してしまふ．

表 7 より実アイテムバンク  $n = 1000$  について， $OC = 10$  の条件では整数計画法は 10 万に近いテストを構成できている．一方で RndMCP 法はいずれの  $OC$  の条件下でも，ほぼテストが構成できていない．この原因を分析するために，RndMCP 法の頂点生成においてカテゴリカル制約のみを課した場合にテストが全て制約を満たす回数を実験で確認した．具体的にはテストの構成回数は  $10^8$  回，制約は C2~C13, C21~C31 の 23 個のカテゴリカル制約のみを課して，これらのうち頂点が全ての制約を満たす回数を実験的に数え上げた．その結果，23 個のカテゴリカル制約を全て満たすテストを構成できたのは  $10^8$  回中 2 回のみであった．このことから，RndMCP 法のランダム探索で本実験のカテゴリカル制約をすべて同時に満たすテストを構成する確率は小さく実験時間  $CT = 8hr$  では十分な数のテストを構成することができなかつたと考えられる．

$n = 2000$  以外の  $OC = 5$  の条件では，テスト構成数が RndMCP 法よりも大きくなった．そこで，テスト構成数の増加の原因について分析するために，最大クリーク探索を行うグラフの頂点集合  $V$  における露出数  $IEV_i$  式 (13) と見つかった最大クリーク  $C$  における露出数  $IEC_i$  式 (16) を用いて，項目  $i$  を含む頂点のうち，最大クリークに選ばれた割合  $CR_i$  を以下のように定義する．

$$CR_i = \frac{IEC_i}{IEV_i} \quad (20)$$

この  $CR_i$  が大きいほど，項目  $i$  を含む頂点が最大クリーク  $C$  に選ばれる割合が高くなる．

図 7 は  $n = 978$ ,  $OC = 5$  の条件下で横軸に露出数  $IEV_i$ ，縦軸に  $CR_i$  をとったグラフである．露出数  $IEV_i$  が大きいほど， $CR_i$  が小さくなる．これは露出数が大きい項目を含む頂点は最大クリークに選ばれる割合が低くなることを示している．したがって，整数計画法を用いて，露出数  $IEV_i$  を抑えたグラフから探索を行うことで，最大クリークに選ばれる頂点の割合を高め，テスト構成数を増加させたと考えられる．また，RndMCP 法と提案手法を比較すると，露出数  $IEV_i$  が同じ値でも  $CR_i$  が大きい．これは提案手法の露出数  $IEV_i$  の最大値が小さくなることで，最大クリークに選ばれる頂点の割合が全体的に高くなることを示している．

比較実験により，第 1 段階目のテスト構成において整数計画法を用いることで，露出率

や露出数の偏りを抑えるだけでなく、テスト構成数も増加する可能性があることが分かった。この提案手法を先行研究 [15, 17] の第 1 段階目に用いることで、テスト構成数の更なる増加を期待できる。

## 5.2 提案手法における $s$ の最適化

前節の通り、本提案手法の整数計画法では項目露出数の一様性が考慮されておらず、特定の項目が過度に出題されてしまう。この問題を解決するために、提案手法では露出数が上位  $s$  位までの項目をアイテムバンクから除外し、残りの項目からテストを生成する。本章ではこのパラメータ値  $s$  を変化させたときの露出率を比較し、露出率が最小となる  $s$  の最適値を決定する。具体的には  $s = \{0, 1, 2, 3\}$  の 4 通りの値で露出率を比較する。なお、 $s = 0$  の場合はアイテムバンクからどの項目も除外しないことを意味し、前節 (5.1) の整数計画法と同等である。

パラメータ値は  $L_1 = 100000$ ,  $L_2 = 4hr$ ,  $CT = 4hr$  とした。ただし、実アイテムバンク  $n = 978,1000$  の場合は計算時間  $CT = 4hr$  以内に頂点を  $L_1$  個生成できなかったため、 $CT = 8hr$  と計算時間を延長した。項目重複数条件は  $OC = \{0, 5, 10\}$  の 3 通りで実験を行った。

結果を表 8 に示す。

表 8  $s$  の最適化実験の結果

Pool Size	OC	$s = 0$			$s = 1$			$s = 2$			$s = 3$		
		$ C $	$IEC_{\max}$	$IEC_{\text{rate}}$	$ C $	$IEC_{\max}$	$IEC_{\text{rate}}$	$ C $	$IEC_{\max}$	$IEC_{\text{rate}}$	$ C $	$IEC_{\max}$	$IEC_{\text{rate}}$
500	0	10	1	10.0%	<b>11</b>	1	<b>9.1%</b>	10	1	10.0%	<b>11</b>	1	<b>9.1%</b>
	5	4952	402	8.1%	4994	379	7.6%	<b>5015</b>	375	<b>7.5%</b>	4936	373	7.6%
	10	<b>99979</b>	7259	7.3%	99970	5370	<b>5.4%</b>	99955	5391	5.4%	99946	5428	5.4%
1000	0	<b>18</b>	1	<b>5.6%</b>	<b>18</b>	1	<b>5.6%</b>	<b>18</b>	1	<b>5.6%</b>	<b>18</b>	1	<b>5.6%</b>
	5	<b>52023</b>	2206	4.2%	50779	1616	<b>3.2%</b>	50645	1674	3.3%	50175	1653	3.3%
	10	<b>100000</b>	4139	4.1%	99998	2847	<b>2.8%</b>	99998	2859	2.9%	99998	2888	2.9%
2000	0	31	1	3.2%	<b>32</b>	1	<b>3.1%</b>	<b>32</b>	1	<b>3.1%</b>	<b>32</b>	1	<b>3.1%</b>
	5	88651	1674	1.9%	97320	1397	<b>1.4%</b>	<b>97380</b>	1398	1.4%	97251	1399	1.4%
	10	90917	1707	1.9%	<b>100000</b>	1418	<b>1.4%</b>	<b>100000</b>	1419	1.4%	<b>100000</b>	1421	1.4%
978 (actual)	0	<b>19</b>	1	<b>5.3%</b>	<b>19</b>	1	<b>5.3%</b>	<b>19</b>	1	<b>5.3%</b>	<b>19</b>	1	<b>5.3%</b>
	5	<b>54964</b>	2303	4.2%	54800	1683	<b>3.1%</b>	54853	1695	3.1%	54565	1677	3.1%
	10	<b>100000</b>	4301	4.3%	<b>100000</b>	2754	<b>2.8%</b>	<b>100000</b>	2760	2.8%	99998	2770	2.8%
1000 (actual)	0	<b>10</b>	1	<b>10.0%</b>	<b>10</b>	1	<b>10.0%</b>	<b>10</b>	1	<b>10.0%</b>	<b>10</b>	1	<b>10.0%</b>
	5	7992	944	<b>11.8%</b>	8016	1040	13.0%	<b>8036</b>	1046	13.0%	8016	1011	12.6%
	10	<b>99989</b>	12685	12.7%	<b>99989</b>	10000	<b>10.0%</b>	99988	10001	10.0%	<b>99989</b>	10001	10.0%



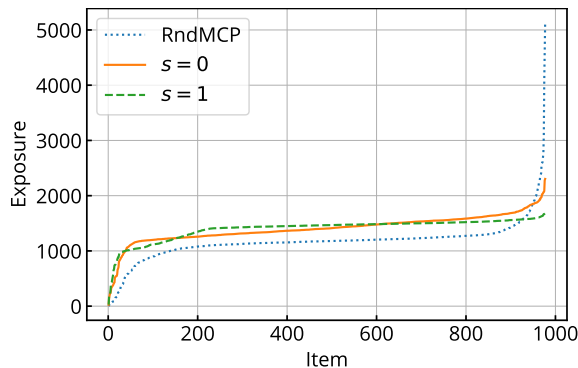


図 8 RndMCP 法と提案手法の露出数の分布 ( $n=978$  (実データ),  $OC=5$ )  
(Copyright ©2022 IEICE [1] 図 8)

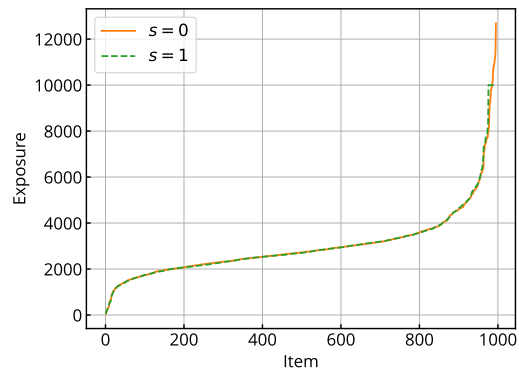


図 9 提案手法の露出数の分布 ( $n=1000$  (実データ),  $OC=10$ )

シミュレーションアイテムバンク  $n = 500, 1000, 2000$  と実アイテムバンク  $n = 978$  において、アイテムバンクから項目を除外しない  $s = 0$  の場合と項目を除外する  $s \geq 1$  の場合を比較すると、すべての  $OC$  の条件において、 $s \geq 1$  の場合のほうが露出率が小さい。これは  $s = 0$  で整数計画法でテストを構成するよりも、 $s \geq 1$  で項目を除外してテストを構成することで特定の項目が過度に出題されないことを示している。さらに、 $s \geq 1$  の露出率を比較すると、 $n = 500, OC = 5$  の条件を除いて  $s = 1$  の場合が最も露出率が小さい。また、表 8 で  $s$  の値が 1 以上の場合は露出率が大きく変化していない。このことは  $s$  が 1 のときに広い範囲で一様に露出数の高い項目を除外できていることを示し、 $s=1$  が有効に機能していることがわかる。図 8 は  $n = 978, OC = 5$  の条件下で RndMCP 法と露出数上位  $s$  位までの項目を除外する提案手法を用いた場合のテスト構成の露出数の分布である。縦軸は露出数、横軸は項目を表し、項目は露出数の昇順に並び替えている。図 8 から、露出数上位 1 位の項目を除外することで特定の項目（右端）が過度に出題される問題が改善されることが確認できる。

表 8 の science データセットの実アイテムバンク  $n = 1000$  において、アイテムバンクから項目を除外しない  $s = 0$  の場合と項目を除外する  $s \geq 1$  の場合を比較すると、 $OC = 10$  の条件では、 $s \geq 1$  の場合のほうが露出率は小さい。図 9 は science データセットの実アイテムバンク  $n = 1000, OC = 10$  の提案手法  $s = 0$  と  $s = 1$  で構成したテストの露出数の分布である。縦軸は露出数、横軸は項目を表し、項目は露出数の昇順に並び替えている。図 9 より、 $OC = 10$  の条件下では実アイテムバンク  $n = 1000$  でも、露出数上位 1 位の項目を除外することで特定の項目（右端）が過度に出題される問題が改善されることが確認できる。

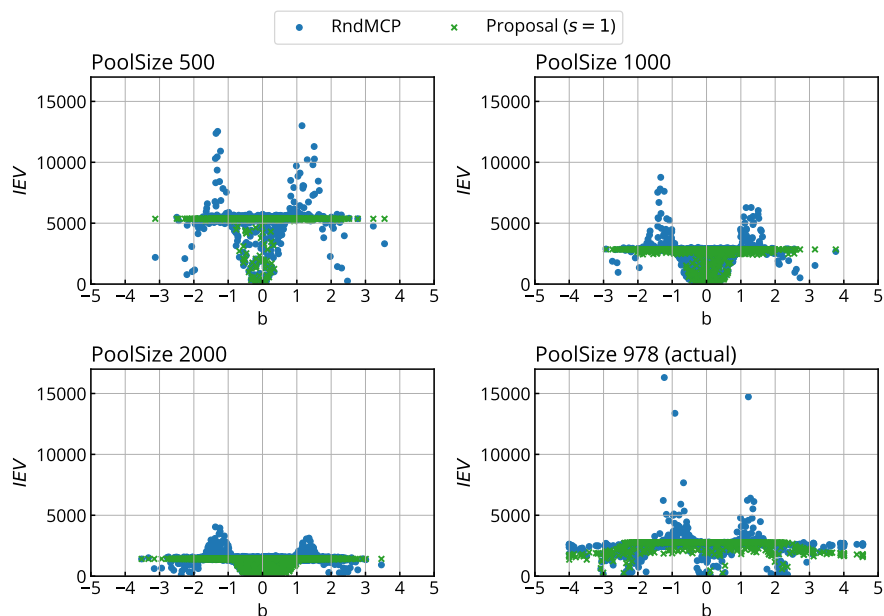


図 10 RndMCP 法と提案手法 ( $s = 1$ ) による項目パラメータ  $b$  と露出数  $IEV$  の比較  
( Copyright ©2022 IEICE [1] 図 9 )

$OC = 10$  の条件下で、提案手法 ( $s \geq 1$ ) を用いても露出率  $IEC_{rate}$  が 10.0% と他のアイテムバンクと比較すると大きいですが、これは science データセットの制約 (表 12) C13 に起因する。C13 はカテゴリ OBJECTIVE が “2A” の項目数を 2 とする制約であり、OBJECTIVE が “2A” の項目はアイテムバンク中に 20 項目ある (表 12 の COUNT 列参照)。これらの項目はテスト構成  $C$  において、2 項目出題されるため、OBJECTIVE が “2A” の項目の露出数の和は  $2|C|$  となり、これらの項目の最大露出数が最小となるのは、20 項目すべてが同じ露出数となるときである。したがって、これらの項目の最大露出数の下限値は  $\frac{2|C|}{20} = \frac{|C|}{10}$  である。露出率は  $IEC_{rate} = IEC_{max}/|C|$  (式 (15)) より、露出率の下限値は  $\frac{|C|}{10}/|C| = 10\%$  である。よって、提案手法 ( $s \geq 1$ ) は実アイテムバンク  $n = 1000$  において  $OC = 10$  の条件下で露出率を下限値付近まで抑制したテストを構成できている。

一方で  $OC = 5$  の条件では、露出率が最小となるのは、 $s = 0$  であった (表 8)。

以降の実験では  $n = 500$ ,  $OC = 5$  の条件で  $s = 2$ ,  $n = 1000$  (実データ),  $OC = 5$  の条件で  $s = 0$ , それ以外の条件では  $s = 1$  を提案手法の  $s$  の最適値とする。

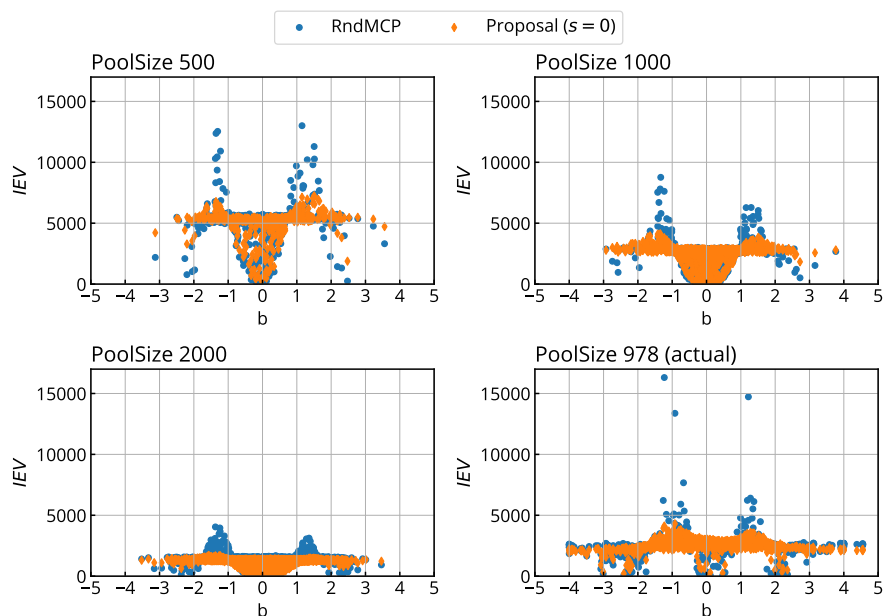


図 11 RndMCP 法と提案手法 ( $s = 0$ ) による項目パラメータ  $b$  と露出数  $IEV$  の比較  
( Copyright ©2022 IEICE [1] 図 10 )

### 5.3 RndMCP 法と提案手法により選択される項目の分析

提案手法により特定の項目が過度に出題される問題が改善された．そこで，RndMCP 法と従来手法で選択される項目特性の差異を分析する．図 10 は  $n = 500, 1000, 2000, 978$  (実アイテムバンク)， $OC = 10$  の条件下で RndMCP 法と提案手法 ( $s = 1$ ) により選択された項目のパラメータ  $b$  を横軸に，露出数  $IEV$  を縦軸にプロットした図である．図 11 は図 10 と同様の条件下で RndMCP 法と提案手法 ( $s = 0$ ) により選択された項目パラメータ  $b$  と露出数をプロットした図である．ただしいずれの図においても  $n = 978$  のアイテムバンクにおいて一度も出題されなかった項目 ( $a = 3.08, b = -0.11$ ) は除外してプロットした． $n = 500, 1000, 978$  のアイテムバンクで， $b \in (-2, -1), b \in (1, 2)$  の項目が RndMCP 法や提案手法 ( $s = 0$ ) では過度に選択されていることがわかる． $b$  の分布は標準正規分布に従うため  $b \in (-2, -1), b \in (1, 2)$  の項目数は  $b \in (-1, 1)$  と比較して少ないが，テスト構成では能力の高い (低い) 少数の受検者の測定精度も保証するため，RndMCP 法は露出数を考慮せず，これらの能力に対応する少数項目を重複して選択していることが理由と考えられる．一方で， $n = 2000$  の比較的大きいアイテムバンクでは露出数の偏りが小さい (図 10, 図 11)．これは  $n = 2000$  では  $b \in (-2, -1), b \in (1, 2)$  の項目数が多いため，露出数

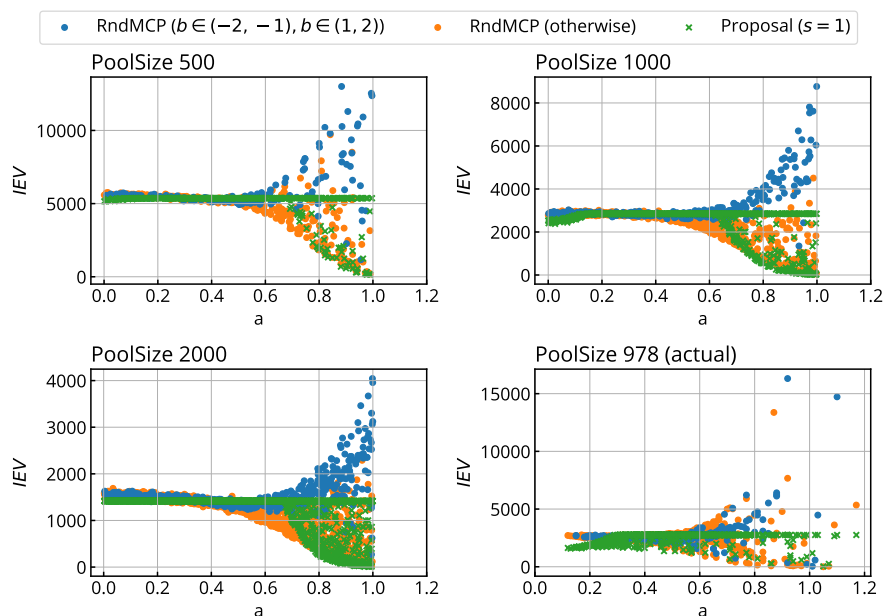


図 12 RndMCP 法と提案手法による項目パラメータ  $a$  と露出数  $IEV$  の比較  
 ( Copyright ©2022 IEICE [1] 図 11 )

を考慮していない RndMCP 法や提案手法の  $s = 0$  のアルゴリズムでも重複せずに項目選択でき、露出数の偏りが小さくなるからと考えられる。図 12 は  $b$  の範囲を項目露出数の大きい  $b \in (-2, -1), b \in (1, 2)$  とそれ以外 ( 図中では「otherwise」) に分けて、パラメータ  $a$  を横軸に、露出数  $IEV$  を縦軸にプロットした図である。 $a$  については RndMCP 法では、 $a$  の値が大きくなるにつれて  $b \in (-2, -1), b \in (1, 2)$  の項目の露出数は大きくなり、それ以外の  $b$  の値の項目の露出数は小さくなる ( 図 12 )。これは、図 10 の分析で示したように  $b$  の範囲に対する項目数の差異が影響していると考えられる。一方で、提案手法は露出数を考慮して項目を選択するため、特定の  $b$  や  $a$  付近の値を持つ少数の項目が過度に出題されるという特性を抑制できている。

次に、提案手法のアルゴリズムにより頂点生成時に除外される項目の特性について分析する。図 13 はアイテムバンク  $n = 978, OC = 10$  の条件下で提案手法 ( $s = 1$ ) で頂点生成した際の項目のパラメータ  $b$  を横軸に、除外した項目の回数を縦軸にプロットした図である。図 13 より提案手法のアルゴリズムでは  $b \in (-2, -1), b \in (1, 2)$  の項目を多く除外して頂点を生成したことがわかる。

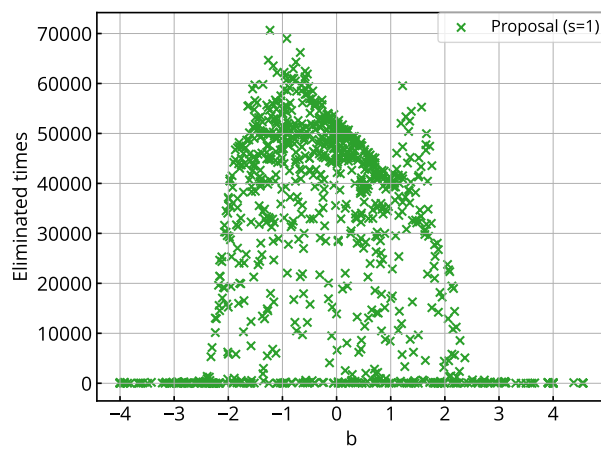


図 13 提案手法により除外された項目パラメータ  $b$  の分布  
 ( Copyright ©2022 IEICE [1] 図 12 )

表 9 第 1 段階目と第 2 段階目を統合したテスト構成の実験結果

Pool Size	OC	RndMCP/MCALIE			RndMCP/HybridRBP			提案手法/MCALIE			提案手法/HybridRBP		
		C	$IEC_{\sigma}$	$IEC_{max}$	$IEC_{rate}$	C	$IEC_{\sigma}$	$IEC_{max}$	$IEC_{rate}$	C	$IEC_{\sigma}$	$IEC_{max}$	$IEC_{rate}$
500	0	<b>18</b>	0.3	1	<b>5.6%</b>	<b>18</b>	0.3	1	<b>5.6%</b>	<b>18</b>	0.3	1	<b>5.6%</b>
	5	11475	130.8	858	7.5%	15795	177.1	1216	7.7%	13430	143.3	998	<b>7.4%</b>
	10	<b>104914</b>	1778.0	13267	12.6%	104817	1777.8	13332	12.7%	99970	1138.4	5370	<b>5.4%</b>
1000	0	<b>36</b>	0.3	1	<b>2.8%</b>	35	0.3	1	2.9%	35	0.3	1	2.9%
	5	55047	492.1	3691	6.7%	55087	492.5	3712	6.7%	50781	377.7	1616	<b>3.2%</b>
	10	<b>104260</b>	1019.9	8762	8.4%	104228	1019.5	8724	8.4%	99998	769.5	2847	<b>2.8%</b>
2000	0	<b>70</b>	0.3	1	<b>1.4%</b>	69	0.3	1	1.4%	<b>70</b>	0.3	1	<b>1.4%</b>
	5	99620	466.5	4028	4.0%	99783	467.0	4031	4.0%	97348	378.0	1397	<b>1.4%</b>
	10	102775	482.4	4105	4.0%	<b>102849</b>	482.6	4106	4.0%	100000	388.7	1418	<b>1.4%</b>
978 (actual)	0	<b>36</b>	0.3	1	<b>2.8%</b>	35	0.3	1	2.9%	<b>36</b>	0.3	1	<b>2.8%</b>
	5	51644	388.7	5342	10.3%	51984	389.8	5357	10.3%	54800	204.6	1683	<b>3.1%</b>
	10	<b>102862</b>	996.1	16698	16.2%	102624	995.7	16686	16.3%	100000	418.7	2754	<b>2.8%</b>
1000 (actual)	0	<b>10</b>	0.5	1	<b>10.0%</b>	<b>10</b>	0.5	1	<b>10.0%</b>	<b>10</b>	0.5	1	<b>10.0%</b>
	5	8432	128.0	1033	12.3%	13978	209.4	1754	12.5%	7996	112.0	944	<b>11.8%</b>
	10	1986	31.1	238	12.0%	20393	312.4	2549	12.5%	99989	1528.1	10000	<b>10.0%</b>

## 5.4 第1段階目と第2段階目を統合したテスト構成の比較実験

第1段階目で提案手法を用いることによる第2段階目のテスト構成のテスト構成数や露出率への影響を確認するために、第1段階目で RndMCP 法 [14] を用いた場合と提案手法を用いた場合で第2段階目の比較実験を行った。第2段階目のテスト構成手法は MCALIE 法 [24] と HybridRBP 法 [16, 15] でテスト構成数及び露出率を比較した。各手法の計算時間は 24hr とし、HybridRBP 法と MCALIE 法のパラメータは  $L_1 = 100,000$  ,  $L_2 = 4hr$  ,  $\alpha = 0.1$  ,  $CT' = 4hr$  (シミュレーションデータ) ,  $CT' = 8hr$  (実データ) ,  $CT = 24hr$  とした。第1段階目の提案手法のテスト構成は前述の比較実験 (5.2) で最も露出率が小さいものを採用した。項目重複数条件は  $OC = \{0, 5, 10\}$  の3通りの条件で比較した。

結果を表9に示す。第1行では第1段階目の手法/第2段階目の手法として表記している。本論文では第1段階目の手法を提案しているため、第2段階目に MCALIE 法を用いた場合と HybridRBP 法を用いた場合に分けて分析する。

### 5.4.1 提案手法/MCALIE 法

表9より、第1段階目に提案手法を用いた MCALIE 法は各手法と比較して多くの条件下で露出率は最小であるが、 $OC = 0$  の場合を除いてテスト構成数は小さい。

この原因を分析するために表10に第1段階目のテスト構成の結果を示し、表11に第2段階目のテスト構成数を示した。表11より、第1段階目に提案手法を用い第2段階目は MCALIE 法を用いた場合は  $OC = 0$  と  $n = 500$  ,  $OC = 5$  の場合を除いておおよそテスト構成数が少ないことがわかる。

さらにこの原因を明らかにするために、図14に  $n = 978$  ,  $OC = 10$  の条件の MCALIE 法におけるテスト構成の露出率の遷移を示す。図14より、提案手法/MCALIE 法の組み合わせでは、第1段階目で構成したテスト構成が構成した複数のテストの中で露出率が最小であり、MCALIE 法がそのテスト構成を選択し続けるため、テスト構成数が 100,000 から増加しないことが分かる。以上をまとめると、提案手法/MCALIE 法の組み合わせは従来手法と比較して露出率は小さくなるが、MCALIE 法において露出率が第1段階目で求めたテスト構成から減少しないため、テスト構成数が少なくなった。

表 10 第 1 段階目のテスト構成の実験結果

Pool Size	OC	RndMCP				提案手法			
		$ C $	$IEC_{\sigma}$	$IEC_{max}$	$IEC_{rate}$	$ C $	$IEC_{\sigma}$	$IEC_{max}$	$IEC_{rate}$
500	0	10	0.5	1	10.0%	<b>11</b>	0.5	1	<b>9.1%</b>
	5	4371	53.1	378	8.6%	<b>5015</b>	50.7	375	<b>7.5%</b>
	10	<b>99981</b>	1729.2	13006	13.0%	99970	1138.4	5370	<b>5.4%</b>
1000	0	17	0.5	1	5.9%	<b>18</b>	0.5	1	<b>5.6%</b>
	5	46190	430.5	3374	7.3%	<b>50779</b>	377.7	1616	<b>3.2%</b>
	10	<b>100000</b>	990.7	8767	8.8%	99998	769.5	2847	<b>2.8%</b>
2000	0	<b>32</b>	0.5	1	<b>3.1%</b>	<b>32</b>	0.5	1	<b>3.1%</b>
	5	96773	456.1	3833	4.0%	<b>97320</b>	377.9	1397	<b>1.4%</b>
	10	<b>100000</b>	472.6	4044	4.0%	<b>100000</b>	388.7	1418	<b>1.4%</b>
978 (actual)	0	18	0.5	1	5.6%	<b>19</b>	0.5	1	<b>5.3%</b>
	5	45790	368.2	5177	11.3%	<b>54800</b>	204.6	1683	<b>3.1%</b>
	10	<b>100000</b>	985.3	16310	16.3%	<b>100000</b>	418.7	2754	<b>2.8%</b>
1000 (actual)	0	0	-	0	-	<b>10</b>	0.5	1	<b>10.0%</b>
	5	0	-	0	-	<b>7992</b>	111.9	944	<b>11.8%</b>
	10	1	0.2	1	100%	<b>99989</b>	1528.1	10000	<b>10.0%</b>

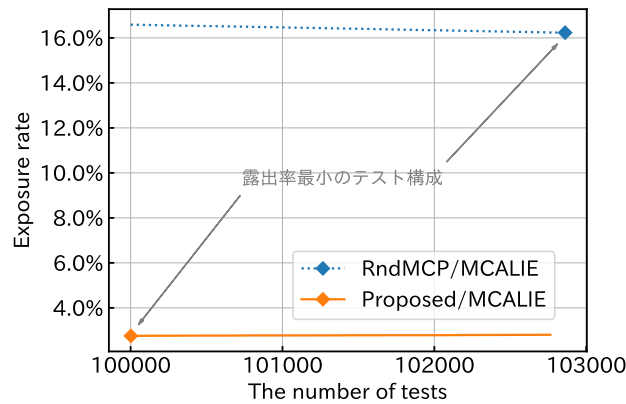


図 14 MCALIE 法における露出率の遷移  
( Copyright ©2022 IEICE [1] 図 13 )

#### 5.4.2 提案手法/HybridRBP 法

表 9 より，実アイテムバンク  $n = 1000$  以外のアイテムバンクで第 1 段階目に提案手法を用いた HybridRBP 法は第 1 段階目に RndMCP 法を用いた手法と比較すると  $OC = 0$  以外のすべての条件で，露出率は小さい．テスト構成数は  $OC = 5$  では大きく， $OC = 0, 10$



表 11 第 2 段階目のテスト構成数

Pool Size	OC	RndMCP/MCALIE	RndMCP/HybridRBP	提案手法/MCALIE	提案手法/HybrdRBP
		C	C	C	C
500	0	<b>8</b>	<b>8</b>	7	6
	5	7104	<b>11424</b>	8415	11001
	10	<b>4933</b>	4836	0	4714
1000	0	<b>19</b>	18	17	16
	5	8857	<b>8897</b>	2	8537
	10	<b>4260</b>	4228	0	4252
2000	0	<b>38</b>	37	<b>38</b>	<b>38</b>
	5	2847	3010	28	2876
	10	2775	<b>2849</b>	0	2839
987 (actual)	0	<b>18</b>	17	17	16
	5	5854	<b>6194</b>	0	4868
	10	<b>2862</b>	2624	0	2765
1000 (actual)	0	<b>10</b>	<b>10</b>	0	0
	5	8432	<b>13978</b>	4	7489
	10	1985	<b>20392</b>	0	1544

ではおおよそ同様である。OC = 5 でテスト構成数が増加したのは、第 1 段階目の提案手法によるテスト構成数が OC = 5 で RndMCP 法よりも増加したからである（表 10）。また表 11 より、OC = 5 では第 2 段階目において第 1 段階目に RndMCP 法を用いたものよりもテスト構成数は少なくなった。これは提案手法では第 1 段階目のテスト構成数が増加したことにより、第 2 段階目の整数計画問題の項目重複数制約（式（12））が多くなり、整数計画問題の求解時間が長くなったからである。

実アイテムバンク  $n = 1000$  で第 1 段階目に提案手法を用いた HybridRBP 法は第 1 段階目に RndMCP 法を用いた HybridRBP 法と比較すると提案手法の OC = 10 の条件では従来手法より露出率は小さく、テスト構成数が大きいテストを構成できた。しかし、OC = 5 の条件下ではテスト構成数は提案手法のほうが大きい、露出率に関してはほぼ同じである。これは従来手法は第 1 段階目の RndMCP 法でテストを構成できず（表 10 参照）、第 2 段階目の整数計画法でテスト構成を行ったため露出数の偏りが生じなかったからである。

以上をまとめると、提案手法/HybridRBP 法の組み合わせは多くの条件下で従来手法と比較してテスト構成数を減らすことなく、露出率が低いテストを構成できることが分かった。

## 6 むすび

本論文では e テスティングにおける，項目露出を制約する並行テスト構成手法の提案と評価を行った．本手法では RndMCP 法のランダム探索による項目露出の偏りを制約を満たす頂点集合から頂点をなるべく一様に生成できるように目的関数に一様分布に従う係数を掛け合わせ最大化させることで改善した．しかし，この整数計画法は項目ごとの露出数の一様性は考慮していないため，依然として特定の項目が過度の出題されてしまう問題がある．そこで，露出数が上位の項目を逐次的にアイテムバンクから除外することで，多くの条件下でこの問題を緩和できた．以上の提案手法の有効性をシュミレーション・実データを用いて示した．

露出率の抑制だけでなく，Fuchimoto et al. (2022) の並列探索アルゴリズム [17, 18] や 湊本，湊，植野 (2022) の ZDD という二分木を圧縮表現可能なデータ構造を用いたテスト構成手法 [19] に本手法を組み込むなどして，テスト構成数を大きくするアルゴリズムの開発が今後の課題である．

更に，構成した弱並行テストを適応型テストに用いることで，テストの長さや項目の露出数を軽減することが知られている．適応型テストとは，受検者の能力を逐次的に推定し，その能力に応じて測定制度が最も高い項目を出題することで，受検時間や項目数を軽減できるコンピュータ・テストの出題方式である．本手法のように，予め露出数を考慮したテスト群を生成することで，テストの長さや露出数を改善できる可能性がある．このような，実用上の課題についても検討の余地がある．

## 謝辞

本論文を作成するにあたり，指導教員の植野真臣教授から，丁寧かつ熱心なご指導を賜りました．ここに感謝の意を表します．また，日常の議論や論文執筆を通じて多くの示唆や知識を頂いた研究室の先輩・同期・後輩に感謝いたします．

## 参考文献

- [1] 植野晶, 淵本壱真, 植野真臣, “項目露出を考慮した整数計画法による等質テスト構成,” 電子情報通信学会論文誌 D, vol.105, no.8, pp.485–498, 2022.
- [2] 植野真臣, 永岡慶三, e テスティング, 培風館, 2009.
- [3] M.Ueno, K.Fuchimoto, and E.Tsutumi, “e-testing from artificial intelligence approach,” Behaviormetrika, vol.48, no.2, pp.409–424, 2021.
- [4] M. Ueno, “Ai based e-testing as a common yardstick for measuring human abilities.in: The 18th international joint conference on computer science and software engineering,” IEEE computer society, pp.1–6, 2021.
- [5] F.M. Lord and M.R. Novick, Statistical theories of mental test scores, IAP, 1968.
- [6] F. Samejima, “Weakly parallel tests in latent trait theory with some criticisms of classical test theory,” Psychometrika, vol.42, no.2, pp.193–198, 1977.
- [7] W.J. van der Linden, Linear Models for Optimal Test Design, Springer, 2005.
- [8] D.I. Belov and R.D. Armstrong, “A constraint programming approach to extract the maximum number of non-overlapping test forms,” Computational Optimization and Applications, vol.33, no.2-3, pp.319–332, 2006.
- [9] D.I. Belov, “Uniform test assembly,” Psychometrika, vol.73, no.1, p.21, 2008.
- [10] K.T. Sun, Y.J. Chen, S.Y. Tsai, and C.F. Cheng, “Creating irt-based parallel test forms using the genetic algorithm method,” Applied measurement in education, vol.21, no.2, pp.141–161, 2008.
- [11] P. Songmuang and M. Ueno, “Bees algorithm for construction of multiple test forms in e-testing,” IEEE Transactions on Learning Technologies, vol.4, no.3, pp.209–221, 2010.
- [12] P. Songmuang and M. Ueno, “Multiple test forms construction based on bees algorithm,” Educational Data Mining 2010, The 3rd International Conference on Educational Data Mining, Pittsburgh, PA, USA, June 11-13, 2010. Proceedings, eds. by R.S.J. deBaker, A. Merceron, and P.I.P. Jr., pp.321–322, www.educationaldatamining.org, 2010.
- [13] T. Ishii, P. Songmuang, and M. Ueno, “Maximum clique algorithm for uniform test forms assembly,” Artificial Intelligence in Education, eds. by H.C. Lane, K. Yacef, J. Mostow, and P. Pavlik, pp.451–462, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [14] T. Ishii, P. Songmuang, and M. Ueno, “Maximum clique algorithm and its approximation for uniform test form assembly,” IEEE Transactions on Learning Technologies, vol.7,

no.1, pp.83–95, 2014.

- [15] 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,” 電子情報通信学会論文誌 D, vol.100, no.1, pp.47–59, 2017.
- [16] T. Ishii and M. Ueno, “Algorithm for uniform test assembly using a maximum clique problem and integer programming,” International Conference on Artificial Intelligence in EducationSpringer, pp.102–112 2017.
- [17] 淵本壱真, 植野真臣, “等質テスト構成における整数計画法を用いた最大クリーク探索の並列化,” 電子情報通信学会論文誌 D, vol.103, no.12, pp.881–893, 2020.
- [18] K. Fuchimoto, T. Ishii, and M. Ueno, “Hybrid maximum clique algorithm using parallel integer programming for uniform test assembly,” IEEE Transactions on Learning Technologies, vol.15, no.2, pp.252–264, 2022.
- [19] 淵本壱真, 湊真一, 植野真臣, “Zero-suppressed binary decision diagrams を用いた自動テスト構成,” 人工知能学会論文誌, vol.37, no.5, pp.A–M23–1–11, 2022.
- [20] 石井隆稔, 植野真臣他, “最大クリーク問題を用いた複数等質テスト自動構成手法,” 電子情報通信学会論文誌 D, vol.97, no.2, pp.270–280, 2014.
- [21] E. Tomita, S. Matsuzaki, A. Nagao, H. Ito, and M. Wakatsuki, “A much faster algorithm for finding a maximum clique with computational experiments,” Journal of Information Processing, vol.25, pp.667–677, 2017.
- [22] C.M. Li, H. Jiang, and F. Many’a, “On minimization of the number of branches in branch-and-bound algorithms for the maximum clique problem,” Computers & Operations Research, vol.84, pp.1–15, 2017.
- [23] H. Wainer, “Cats: Whither and whence,” *Psicologica*, vol.21, no.1, pp.121–133, 2000.
- [24] T. Ishii and M. Ueno, “Clique algorithm to minimize item exposure for uniform test forms assembly,” International Conference on Artificial Intelligence in EducationSpringer, pp.638–641 2015.
- [25] 植野晶, “項目露出を考慮した整数計画法による等質テスト構成,” 卒業論文, 電気通信大学, 2021.
- [26] S.W. Choi, S. Lim, and W.J. van derLinden, “Testdesign: an optimal test design approach to constructing fixed and adaptive tests in r,” *Behaviormetrika*, vol.49, no.2, pp.191–229, 2022.
- [27] E. Boekkooi-Timminga, “The construction of parallel tests from irt-based item banks,” *Journal of Educational Statistics*, vol.15, no.2, pp.129–145, 1990.

- [28] R.D. Armstrong, D.H. Jones, and Z. Wang, "Automated parallel test construction using classical test theory," *Journal of Educational Statistics*, vol.19, no.1, pp.73–90, 1994.
- [29] W.J. van der Linden and J.J. Adema, "Simultaneous assembly of multiple test forms," *Journal of educational measurement*, vol.35, no.3, pp.185–198, 1998.
- [30] F.M. Lord and M.R. Novick, *Statistical theories of mental test scores*, IAP, 2008.
- [31] F.B. Baker and S.H. Kim, *Item response theory: Parameter estimation techniques*, CRC Press, 2004.
- [32] E. Muraki, "A generalized partial credit model: Application of an em algorithm," *ETS Research Report Series*, vol.1992, no.1, pp.i–30, 1992.
- [33] R.D. Armstrong, D.H. Jones, and C.S. Kuncze, "Irt test assembly using network-flow programming," *Applied Psychological Measurement*, vol.22, no.3, pp.237–247, 1998.
- [34] IBM, "Ilog cplex optimization studio cplex user's manual 12.9," 2019.

表 12 science データセットの制約条件

CONSTRAINT_ID	TYPE	CONDITION	LB	UB	COUNT
C1	NUMBER		30	30	1000
C2	NUMBER	LEVEL = 3	10	10	342
C3	NUMBER	LEVEL = 4	10	10	335
C4	NUMBER	LEVEL = 5	10	10	323
C5	NUMBER	STANDARD = 1	17	20	689
C6	NUMBER	STANDARD $\in$ {2, 4}	6	8	141
C7	NUMBER	STANDARD = 3	2	4	170
C8	NUMBER	OBJECTIVE = 1A	2	3	83
C9	NUMBER	OBJECTIVE $\in$ {1B, 1C, 1I, 1G}	5	6	196
C10	NUMBER	OBJECTIVE $\in$ {1D, 1F}	5	6	228
C11	NUMBER	OBJECTIVE $\in$ {1E, 1J, 1K}	3	4	141
C12	NUMBER	OBJECTIVE = 1H	1	1	30
C13	NUMBER	OBJECTIVE = 2A	2	2	20
C14	NUMBER	OBJECTIVE $\in$ {2B, 2C, 2D}	1	1	34
C15	NUMBER	OBJECTIVE $\in$ {4A, 4D}	1	1	39
C16	NUMBER	OBJECTIVE $\in$ {4B, 4E}	1	1	21
C17	NUMBER	OBJECTIVE $\in$ {4C, 4F}	1	1	25
C18	NUMBER	OBJECTIVE $\in$ {3A, 3D}	3	3	64
C19	NUMBER	OBJECTIVE $\in$ {3B, 3E}	2	3	64
C20	NUMBER	OBJECTIVE $\in$ {3C, 3F}	0	3	43
C21	NUMBER	STANDARD = 1 $\wedge$ DOK $\geq$ 2	7	30	381
C22	NUMBER	STANDARD $\in$ {2, 4} $\wedge$ DOK $\geq$ 3	2	30	42
C23	NUMBER	STANDARD = 3 $\wedge$ DOK $\geq$ 3	2	30	97
C24	NUMBER	TYPE = DRAG	2	4	70
C25	NUMBER	TYPE = EQTN	12	15	449
C26	NUMBER	TYPE = FILL	1	2	21
C27	NUMBER	TYPE = GRAPH	1	3	34
C28	NUMBER	TYPE = HOTS	1	3	72
C29	NUMBER	TYPE = MATCH	2	4	117
C30	NUMBER	TYPE = SRMU	1	2	32
C31	NUMBER	TYPE = SRSI	5	8	205
C32	ORDER	LEVEL			
C33	ENEMY	ID $\in$ {SC00001, SC00002}			2
C34	INCLUDE	ID $\in$ {SC00003, SC00004}			2
C35	EXCLUDE	PTBIS < 0.15			8
C36	ALLORNONE	ID $\in$ {SC00005, SC00006}			2