

測定精度の等質性を向上させる
最大重みクリーク探索と混合整数計画法を用いた
自動テスト構成

2024年2月2日

情報数理工学プログラム

学籍番号 1910172

門脇 瑞穂

指導教員 植野 真臣

令和5年度 情報数理工学プログラム卒業論文概要

| | |
|------------|---|
| 令和1年度 入学 | 学籍番号 1910172 |
| 指導教員 植野 真臣 | 氏名 門脇 瑞穂 |
| 題目 | 測定精度の等質性を向上させる最大重みクリーク探索と混合整数計画法を用いた自動テスト構成 |

概要

e-Testingでは異なる問題で構成されるが測定精度が等質なテスト(並行テスト)が多数必要となる。現在最も多くの並行テストを生成できる手法は頂点をテストとし、テスト間重複項目制約を満たす場合に二頂点間に辺を加える最大クリーク法と整数計画法を用いた並列探索手法である。しかしこの手法では、測定精度の等質性が人手で設定される上限・下限に依存する。テスト構成数の最大化と測定精度の等質化は自動テスト構成における重要な課題であるが、両者にはトレードオフの関係がある。提案手法は最大重みクリーク探索と混合整数計画法を用いてその両方を改善する。具体的には、各頂点の重みとして目標測定精度との差異をペナルティに設定するグラフ構造を考え、最大重みクリーク探索を行う。また、クリーク探索を行う候補頂点集合の決定時にも混合整数計画法を用い、測定精度が目標値に近いものを選ぶ。しかし、最大重みクリーク探索は、クリーク数最大化を目的としていないのでテスト構成数の最大化が保証されない。また、混合整数計画問題は整数計画問題に比べ一般に計算時間が大きくなるため、テスト構成数が増やせない。この問題を解決するために、時間計算量を $O(2^n)$ から $O(2^{rn})$ ($0 < r < 1$)へ減じて混合整数計画問題を解くアルゴリズムを提案する。具体的には、各解の探索空間としてのアイテムバンクを項目全体から確率 r でランダムサンプリングし混合整数計画問題に適用して探索を繰り返すことにより計算量を削減する。実験により、提案手法はテスト構成数を3倍にし、測定精度の散らばりを約100分の1に抑えることが示された。

1 まえがき

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 CBT のことである [1, 2]. e-Testing を用いると、同一能力の受検者は異なるテストを受検しても同一得点が保証される。これにより、受検者は同一精度で複数回の受検を行うことができる。我が国においても医療系大学間共用試験 [3] や情報処理技術者試験 [4] 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。

歴史的には、e-Testing のアイデアは Lord and Novick [5] が同一能力の受検者に対して真に同じ得点を測定する 2 つのテストを並行テストと定義したことから始まる。しかし、この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) [6] は項目反応理論 (Item Response Theory: IRT) [7] 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 [8–15]。以降、本論文ではこの弱並行テストを単に並行テスト、テストと呼ぶ。 [13]

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。並行テストの自動構成については、互いに測定精度が等質な異なる項目の組合せが列挙される。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。

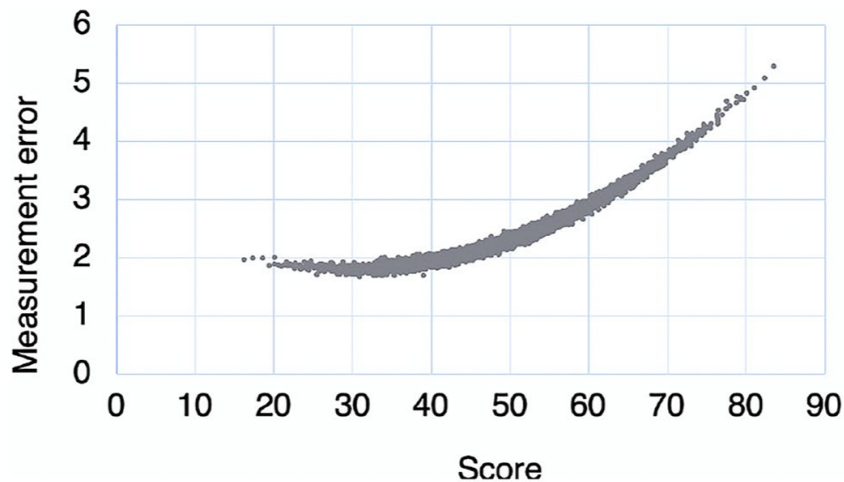


図1 並行テストの得点とその測定誤差の分布

図1はある公的試験で実施された並行テストの得点とその精度(誤差)の分布である。この試験では、各受検者に異なるテストが割り当てられ、カンニング防止や複数日程での受検が可能となっている。図の各プロットは異なるテストの測定結果である。この結果から、実際に同一能力(同一テスト得点)においては、異なるテスト間で測定精度(誤差)が同程度となっていることがわかる。このようなテストの実現において最も重要な研究課題は以下の2点である。

- (1) テスト構成数の最大化
- (2) テスト間での測定精度の等質化

(1)のテスト構成数に関し、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)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)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[16]。また、項目の作問には時間的・金銭的成本がかかる。したがって、自動テスト構成では限られた項目から最大数のテストを構成する手法が求められ

る。(2)の測定精度の等質化とは、図1の各能力値(テスト得点)においてプロットの縦方向の散らばりを小さくすることに相当する。測定精度を等質化することにより、異なるテストの得点が平等に比較可能となる。等質化の方法により、既存手法は以下の2種類に分けられる。

(2.a) 目標値設定型

(2.b) 上限・下限制約型

(2.a)は、測定精度に目標値を設定し、その値に近づくようテスト構成を行う手法である[12]。構成されるテストの測定精度は目標値近辺で等質となる。(2.b)は、測定精度に許容値を設定し、その範囲内であれば等質であるとみなしてテスト構成を行うものである[14,15,17-19]。ここで、並行テストの構成において(1)構成数の最大化と(2)測定精度の等質化はトレードオフの関係にある。それぞれ、(2.a)は(2)等質性のみを各テストについて貪欲的に最適化しており、(2.b)は(1)構成数のみを並行テスト全体について最適化している。ただし、(2.b)は上限・下限の幅を手動で狭めることにより(2)の等質化についても調整を行う。

(2.a)の自動テスト構成手法として最も有名なBig Shadow Test method(BST法)は、混合整数計画法を用い、目標とする測定精度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[12]。そのため、初期段階では(2)測定精度の等質性が非常に高いテストが構成される。しかし、構成数が増えるにつれテストの測定精度が目標値から乖離していく。結果として、構成された並行テストは測定精度の散らばりが大きくなってしまふ。これは、並行テスト全体に対して(2)の最適化が行われていないためであり、BST法に(2.b)の上限・下限制約を加えても同様の結果となる。また、この手法は(1)テスト構成数に関して最適化を行わない。したがって、等質性を保ちながら実用に必要なテスト数を確保することが困難である。

この問題を解決するために、Ishii et al.[17]は、(2.b)として測定精度の等質条件に関しては手動で許容値を設定し、その条件下で構成できるテスト数を最大化する手法を提案した。この手法は自動テスト構成をグラフ上で定義し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(頂点)間に共通する

項目の数が一定数以下である場合に辺を引いたグラフから、クリークと呼ばれる任意の二頂点が隣接するグラフ構造を探索することで並行テストを構成する。

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. [14] 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 [20,21] を用いても、少なくとも $O(|V|^2)$ ($|V|$ はグラフの頂点数) の空間計算量を必要とするため、(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 and Ueno [18] 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 はアイテムバンクサイズ) と大きく、テスト構成数の改善は僅かなものであった [22]。

整数計画法の計算時間を改善するために、Fuchimoto et al. [15] 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 [15]。さらに、並列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 [15]。

しかし、これらの (2.b) の手法 [14,15,17,18] では (2) のテスト間の測定精度の等質性は、手動で設定する上限・下限制約に依存しており最適化されていない。また、(2.a) の BST 法 [12] も異なる並行テスト間における測定精度のちらばり

が大きくなる他、構成数も十分に増やすことができない。この結果は、BST 法に (2.b) の手法の上限・下限を加えても同じである。この問題に対し、本論文では最大重みクリーク探索と、混合整数計画法を用いた近似解探索によって (1) テスト構成数と (2) 等質性の両方を改善する手法を提案する。具体的には、頂点に (2.b) の上限・下限制約を満たすテスト、辺にテスト間重複項目制約への適否、各頂点の重みに目標とする測定精度との差異のペナルティを設定するグラフ構造を考え、最大重みクリーク探索を行う。また、提案手法は現在のクリークとの隣接頂点の並列探索に Linden ら [12] の手法と同様の混合整数計画法を用いる。具体的には、目標とする測定精度からの差異を最小化するように隣接頂点（テスト）を構成する。そして、集めた頂点集合から最大重みクリークを探索し、解を現在のクリークに追加する。この繰り返しによる (2) 等質性の最適化では、Linden らの手法よりも貪欲探索が緩和され、大局的に最適な解を求めることができる。しかし、最大重みクリーク探索は、クリーク数最大化を目的としていないのでテスト構成数の最大化が保証されない。また、混合整数計画問題は整数計画問題に比べ一般に計算時間が大きくなるため、テスト構成数が増やせない。この問題を解決するために、時間計算量を $O(2^n)$ から $O(2^{rn})$ ($0 < r < 1$) へ減じて混合整数計画問題を解くアルゴリズムを提案する。具体的には、各解の探索空間としてのアイテムバンクを項目全体から確率 r でランダムサンプリングし混合整数計画問題に適用して探索を繰り返すことにより計算量を削減する。本提案手法の二段階目において HMAPIPI 法の約 13 倍、全体を通して HMAPIPI 法の約 3 倍のテスト構成を実現した。

本論文では、提案手法の有効性をシミュレーションデータと実データを用いて示した。既存手法と比べ構成数を増加させながら、測定精度のちらばりを小さくすることができた。

2 項目反応理論

項目反応理論 (Item Response Theory:IRT) [5, 23] と呼ばれる数理モデルを用いると、並行テストは、「各テストでの受験者得点の予測誤差が等質である」テ

スト集合として定義される（例えば, [14,17,24]）。ここで, 受験者得点の予測誤差はテストの自動構成に関する研究（例えば, [8-13]）では, IRT におけるテスト情報量で評価されている。

IRT とは受験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, 異なる項目から構成されるテストを受けた受験者の能力を同一尺度上で評価できる。

IRT では, 項目 $i(= 1, 2, \dots, n)$ に対する受験者 $j(= 1, 2, \dots, m)$ の反応 u_{ij} を以下のように表す。

$$u_{i,j} = \begin{cases} 1 & \text{項目 } i \text{ に受験者 } j \text{ が正答} \\ 0 & \text{それ以外} \end{cases}$$

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 $\theta_j \in (-\infty, \infty)$ を持つ受験者 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。

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

ここで, $a_i \in [0, \infty], b_i \in [0, \infty]$ 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。

IRT では項目 i において, 式 (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, 以下のように定義する。

$$I_i(\theta) = 1.7^2 a_i^2 p_i(\theta)(1 - p_i(\theta)) \quad (2)$$

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。

$$I(\theta) = \sum_{i \in U} I_i(\theta) \quad (3)$$

ここで、 U はテストを構成する項目の集合である。このテスト情報量の逆数が受験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 [25]。

ただし、多くの自動テスト構成手法 [8–13] ではテスト情報量における受験者の能力パラメータを $\theta_k (k = 1, 2, \dots, K)$ 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。

3 並行テストの自動構成アルゴリズム

本節では提案手法と関連がある手法を紹介する。

3.1 Big Shadow Test

Linden [12] は、混合整数計画法を用いた Big Shadow Test 法 (BST 法) を提案した。この手法は以下の混合整数計画問題を解くことで、構成するテストと将来構成するテストの候補群 (“shadow test” と呼ぶ) の分割を行う。この分割を逐次的に繰り返すことで並行テストを生成する。

variables

$$y \geq 0$$

$$x_{i,j} = \begin{cases} 1 & \text{項目 } i \text{ がテストに含まれる} \\ 0 & \text{それ以外} \end{cases}$$

$$z_{i,j} = \begin{cases} 1 & \text{項目 } i \text{ が } \textit{shadowtest} \text{ に含まれる} \\ 0 & \text{それ以外} \end{cases}$$

maximize

$$y$$

subject to

$$\sum_{k=1}^K \left| \sum_{i=1}^n I_i(\theta_k) x_i - T(\theta_k) \right| \leq My$$

$$\sum_{k=1}^K \left| \sum_{i=1}^n I_i(\theta_k) z_i - T_{ST}(\theta_k) \right| \leq M_{ST}y$$

$$\sum_{i=1}^n x_i = M$$

$$\sum_{i=1}^n z_i = M_{ST}$$

$$x_i + z_i \leq 1 \quad (i = 1, \dots, n)$$

where

$$T_{ST}(\theta_k) = \frac{M_{ST}}{M} T(\theta_k)$$

M と M_{ST} は構成するテストと shadow test の項目数である。また、 $T(\theta_k)$ は θ_k におけるテスト情報量の目標値であり、 T_{ST} はテスト情報量の目標値に構成するテストと shadow test の項目数の比を掛けたものである。この混合整数計画問題は目的関数 y によって、テスト情報量の目標値と構成するテスト及び shadow test のテスト情報量との差が最小となるように最適化される。したがって、BST 法ではこの問題を繰り返し解くことで、目標のテスト情報量付近の値を持つ並行テスト群が構成される。

3.2 並行テストのための最大クリーク問題

石井ら [14] 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、厳密に最大数の並行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二頂点が隣接しているグラフ構造である。したがって、生成されるテスト候補を以下のグラフ構造とみなし、この中から最大クリークの探索・抽出を行うことで、テストを構成する。

(頂点) アイテムバンクの項目から成るテスト構成条件 (テスト項目数やテスト情報量上限・下限制約 (式 3.2) を満たすテスト。

(辺) 二つのテスト候補で重複する項目数が一定値以下となる場合 (以降, この制約を OC と呼ぶ.), その二つの頂点 (テスト) 間に辺を引く。

BST 法では目標値からの差が最小となるテストを貪欲的に生成するため, テスト構成数の増加にともないテスト間の等質性が失われる問題がある。この問題に対し, 本手法は, 各能力値パラメータ θ_k についてテスト情報量の上限 UB_{θ_k} ・下限 LB_{θ_k} 制約を満たすテストを等質とみなし, 構成数の最適化に焦点を置いている。

$$LB_{\theta_k} \leq I(\theta_k) \leq UB_{\theta_k} \quad (3)$$

3.3 乱択法

前述の, 厳密に最大クリーク問題解く手法ではアイテムバンクサイズが大きくなると, グラフ頂点 (テスト) は組合せ爆発的に増えていく。従って, クリーク探索に必要な時間・空間計算量はそれぞれ $O(2^{|V|})$, $O(|V|^2)$ であるため, 実規模のアイテムバンクから並行テストを構成することが困難になる。

これらの計算コスト緩和するため, 石井ら [ishii2014maximumIEEE, ishii2017AIED] は最大クリーク探索を行う近似アルゴリズムを提案した (以降, RndMCP 法と呼ぶ)。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, ここから最大クリーク探索を繰り返すことで, グラフ全体の最大クリークを近似的に探索する。具体的には Algorithm 1 により, テスト構成を行う。Step1~2 ではテスト構成条件の項目重複数制約以外を満たす L_1 個の頂

点（テスト）を持つテスト候補グラフの部分グラフをランダムに抽出する。ただし、 L_1 はチューニングパラメータであり、メモリ上に保持できる頂点数の上限を計算機環境に合わせて設定する。Step3 では抽出した部分グラフの最大クリーク探索を計算時間 L_2 だけ行う。Step1~3 を計算時間 CT を過ぎるまで繰り返し、Step2 で得られた部分グラフの最大クリークの中から最大のものを出力とする。

Algorithm 1 RndMCP

```

procedure RndMCP( $L_1, L_2, CT$ )
   $C \leftarrow \emptyset, C_{max} \leftarrow \emptyset$ 
   $ST \leftarrow$  current time
  while (current time -  $ST$ ) <  $CT$  do
    /* Step1 */
     $V \leftarrow$  Assemble feasible  $L_1$  tests randomly.
    /* Step2 */
     $G \leftarrow (V, E)$  Generate a graph that corresponds to a set of feasible tests with overlapping items.
    /* Step3 */
     $C \leftarrow$  MCA( $G, L_2$ )
    ▷ MCA( $G, L_2$ ) extracts the maximum clique from the graph  $G$  within calculation time  $L_2$  using Nakanishi and Tomita's maximum clique algorithm [?].
    if  $|C_{max}| < |C|$  then
       $C_{max} \leftarrow C$ 
    end if
  end while
  Output  $C_{max}$ 
end procedure

```

3.4 整数計画法を用いた最大クリークアルゴリズムとその並列化

現在最も多くのテストを生成できる手法として、Fuchimoto ら [15] は最大クリーク問題と整数計画法を用いた二段階並列探索手法を提案した（以降、HMCAPIP 法と呼ぶ）。

一段階目では、比較的計算時間が小さく、空間計算量の大きい RndMCP 法 [14, 18] を用いて、メモリが許す限り多くのテストを構成する。

二段階目では、探索中のクリークの全頂点と隣接する頂点集合を候補頂点集合として、逐次的に整数計画法の解を追加し、定められた要素数となるまで繰り返す。この操作は整数計画法の制約条件を変更せずに行えるため、並列化できる。

ただし，候補頂点集合中の要素は探索中のクリーク的全頂点と隣接しているが，それら自身が互いに隣接している保証はない．そのため，候補頂点集合の中から最大クリークを抽出し，これを探索中のクリークに追加する．これにより，従来手法よりも多くのテストを構成することができた．以下に整数計画問題への定式化を示す．

variables

$$x_i = \begin{cases} 1 & \text{項目 } i \text{ がテストに含まれる} \\ 0 & \text{それ以外} \end{cases}$$

maximize

$$\sum_{i=1}^n \lambda_i x_i$$

subject to

$$\sum_{i=1}^n x_i = L \quad (\text{テストに含まれる項目数})$$

$$L_{\theta_k} \leq \sum_{i=1}^n I_i(\theta_k) x_i \leq U_{\theta_k} \quad (\text{テスト情報量制約})$$

$$(k = 1, \dots, K)$$

$$\sum_{i=1}^n X_{i,r} x_i \leq OC \quad (\text{項目重複数制約})$$

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

where

$$X_{i,m} = \begin{cases} 1 & \text{現在のクリーク } C \text{ のテスト } m \text{ に項目 } i \text{ が含まれる} \\ 0 & \text{それ以外} \end{cases}$$

ここで， $\lambda_i (i = 1, 2, \dots, n)$ は互いに独立な $[0, 1)$ の連続一様分布からの乱数であり，本問題が解かれるたびにリサンプリングされるものとする．制約条件はクリーク C の全頂点と隣接するための条件である．また，目的関数に含まれる λ_i

は項目 x_i に対する重み付けであり、毎回ランダム組み合わせのテストが構成される。すなわち、 λ_i は項目 x_i の優先順位と捉えることもできる。この定式化は Belov [26] で用いられたランダムにテスト構成を行う整数計画法への定式化を項目重複数条件について一般化したものである。

4 最大重みクリーク探索を用いた自動テスト構成手法

HMCAPIP 法 [15] は、測定精度に関し、恣意的に決めたテスト情報量の上限・下限 (式 3.2) を等質条件として用いる。テスト構成数の最大化と測定精度の等質化はトレードオフの関係にあり、手動で調整した等質条件は構成するテスト数に対する最適値ではない。また、整数計画問題を用いた隣接頂点の探索については、計算時間を改善するために並列化をしているものの、解いている問題自体の時間計算量は $O(2^n)$ と大きい。したがってこの時間計算量を改善することで、制限時間内により多くのテスト構成を行うことができる。提案手法は、最大重みクリーク探索及び、ランダムサンプリングを用いる混合整数計画法により、測定精度の等質化とより効率的な構成数の最大化の 2 点を行う。

4.1 最大重みクリーク探索

提案手法は、テスト構成を最大重みクリーク問題に定式化する。具体的には、最大クリーク探索法 [14] と同様に頂点をテスト項目数と上限・下限制約 (式 3.2 を満たすテスト、辺をテスト間重複項目としたグラフを考える。さらに、Linden [12] らの手法のようにテスト情報量の目標値 T_{θ_k} を考え、その目標値 T_{θ_k} との差をペナルティとする重みを各頂点に設定する。このグラフから最大重みクリーク探索を行う。

最大重みクリーク探索とは、無向グラフ $G = (V, E)$ を頂点の有限集合 V 、辺の集合を E 、各頂点 $v (v \in V)$ の重みを $w(v)$ としたとき、以下の式で示される。

variables

$$C \subseteq V$$

maximize

$$\sum_{v \in C} w(v)$$

subject to

$$\forall v', \forall v'' \in C, \{v', v''\} \in E$$

ここで、各頂点 $t (= 1, 2, \dots, |V|)$ について、各頂点の重み $w_t(v_t)$ をテスト情報量の目標値 T_{θ_k} からの差の絶対値の総和 d_t をペナルティとする重みを計算する。

$$d_t = \sum_{k=1}^K (|\sum_{i \in U} I_i(\theta_k) - T(\theta_k)|) \quad (4)$$

$$w_t = -d_t \quad (5)$$

目標値 T_{θ_k} からの差が小さいほど、頂点の重み w_t は大きな値をとる。したがって、最大重みクリークにより探索目標値近辺のテストが優先的に集積される。これにより、構成するテストの情報量のちらばりを抑え、測定精度を等質化することができる。最大重みクリーク探索の手法は数多く提案（例えば、[27, 28]）されているが、本研究では、疎な大規模グラフに対して、効率的に頂点重み付き最大クリークを見つけられる FastWClq 法 [27] を用いる。

HMACAPIP 法 [15] と同様、一段階目において、上記の最大重みクリーク探索をメモリが許す限り行う。また、二段階目における候補頂点集合からのクリーク探索に関しても上記の最大重みクリーク探索を行う。

4.2 混合整数計画問題

提案手法は、候補頂点集合の並列探索における各探索で、以下の 2 点を行う混合整数計画問題を解く。

(1) 測定精度の目標値からの差異の最小化

(2) アイテムバンクの部分集合を用いた近似解探索

(1) により、構成されるテストの測定精度が目標値近辺で等質になる。また、(2)

は、項目全体の集合であるアイテムバンクから確率 r で項目をランダムサンプリングした部分集合を考える。この部分集合から項目の組合せを探索することで、(1) の近似解を得る。(2) により決定変数が減り、解空間の次元が削減されているため混合整数計画問題 1 問あたりの計算時間が短くなる。したがって、制限時間内におけるテスト構成数を増やすことができる。具体的には、厳密探索の時間計算量 $O(2^n)$ を $O(2^{rn})$ ($0 < r < 1$) に減じる。本論文では、この混合整数計画問題を RndMIP と呼ぶ。

二段階目で解く混合整数計画問題は以下のように定式化される。

variables

$$x_i = \begin{cases} 1 & \text{項目 } i \text{ がテストに含まれる} \\ 0 & \text{それ以外} \end{cases}$$

y

minimize

$$y \geq 0$$

subject to

$$L_{\theta_k} \leq \sum_{i=1}^{rn} I_i(\theta_k) x_i \leq U_{\theta_k} \quad (\text{テスト情報量制約 1})$$

$$(k = 1, \dots, K)$$

$$T(\theta_k) - y \leq \sum_{i=1}^{rn} I_i(\theta_k) x_i \leq T(\theta_k) + y \quad (\text{テスト情報量制約 2})$$

$$(k = 1, \dots, K)$$

$$\sum_{i=1}^{rn} x_i = L \quad (\text{テストに含まれる項目数})$$

$$\sum_{i=1}^{rn} X_{i,m} x_i \leq OC \quad (\text{項目重複数制約})$$

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

$$x_i = 0 \quad (\text{確率 } (1 - r) \text{ で決定変数を削減})$$

$$(i = 1, \dots, n)$$

where

$$X_{i,m} = \begin{cases} 1 & \text{現在のクリーク } C \text{ のテスト } m \text{ に項目 } i \text{ が含まれる} \\ 0 & \text{それ以外} \end{cases}$$

y は目標とするテスト情報量からの差の最大値を示す（以降、最大許容誤差と呼ぶ）。 y によって、テスト情報量の目標値と構成するテストのテスト情報量との差が最小となるように最適化される。

しかし、所望のテストの項目重複数制約が厳しい場合、提案する混合整数計画問題ではテスト構成数が増えにくくなる問題がある。これに対し、隣接頂点集合の探索において、上記の混合整数計画問題を用いた探索（RndMIP 法）と従来の整数計画問題に (2) の決定変数の削減を組み合わせた探索（以降、RndIP 法と呼ぶ）を比率 $retio_{MIP} : (1 - retio_{MIP})$ で混合する手法を提案する。具体的には、毎回の探索において確率 $retio_{MIP}$ で RndMIP 法を用い、 $(1 - retio_{MIP})$ で RndIP 法を用いる。本論文では、この手法を RMRIM 法（RndMIP and RndIP Mix Method）と呼ぶ。

5 評価実験

提案手法の有効性を示すため、以下の 4 つの比較実験を行う。

- (1) 一段階目の最大重みクリーク探索と最大クリーク探索の比較
- (2) 二段階目の最大重みクリーク探索と最大クリーク探索の比較
- (3) RndMIP 法と従来手法の比較
- (4) RMRIM 法と従来手法の比較

(1) ではクリーク探索のみの結果を比較する。続いて (2) で二段階目のクリーク

探索において最大重みクリーク探索を行う場合と従来の最大クリーク探索を行う場合の比較を行う。

実験にはシュミレーションおよび実データアイテムバンクを用いた。シュミレーションアイテムバンクは 1000 の項目を持ち、各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 $b \sim N(0, 1^2)$ として発生させた。また、実データは 978 の項目を持ち、パラメータの詳細は [15] を参照されたい。テストの構成条件は、表 1 のテスト情報量制約を満たす 50 項目のテスト構成とした。目標値 $T(\theta_k)$ は、上限・下限内のテストの情報量の中央値を用いた。項目重複数制約 (OC) は、一段階目の実験については 1~20、二段階目の実験については 4, 6, 8, 10, 12, 14, 16, 18, 20 と変化させ比較した。

表 1 50 項目テスト構成のためのテスト情報量制約

| | $\theta = -2.0$ | $\theta = -1.0$ | $\theta = 0.0$ | $\theta = 1.0$ | $\theta = 2.0$ |
|----------------|-----------------|-----------------|----------------|----------------|----------------|
| $UB(\theta_k)$ | 4.8 | 7.2 | 7.2 | 7.2 | 4.8 |
| $LB(\theta_k)$ | 4.0 | 6.4 | 6.4 | 6.4 | 4.0 |

なお、本論文での計算機環境は Ubuntu 18.04.2 を OS とする計算機 (CPU: Intel Core i9-9900X 3.50 GHz, RAM: 128GB) である。

5.1 一段階目における最大重みクリーク探索の評価

各手法の計算時間は最大 6hr [15] とした。その他のパラメータも HMCAPIP 法 [15] と同様である。

はじめに、各手法のテスト構成数 (No.tests) とテスト情報量の標準偏差の θ_k における平均値 ($Avg.SD_{TI}$) を表 2 に示す。 $Avg.SD_{TI}$ は以下の式で表される。

$$Avg.SD_{TI} = \frac{1}{K} \sum_{\theta_k} \sqrt{\frac{1}{No.tests} \sum_{t=1}^{No.tests} (I_t(\bar{\theta}_k) - I_t(\theta_k))^2} \quad (6)$$

提案手法は従来手法と比較して、テスト構成数を従来手法と同程度に保つことができた。また、提案手法はいずれの場合も、テスト情報量のちらばりを小さく抑えることができた。

表 2 テスト構成数とテスト情報量の標準偏差の比較

| Test Length | Item Bank Size | No. tests | | | Avg. SD_{TI} | |
|-------------|----------------|---------------|----------------|----------------|----------------|-----------------|
| | | OC | HMCAPIP | Proposal | HMCAPIP | Proposal |
| | 1000 | 1 | 21 | 21 | 0.14843 | 0.13240 |
| | | 2 | 56 | 57 | 0.16955 | 0.15683 |
| | | 3 | 183 | 192 | 0.17508 | 0.13772 |
| | | 4 | 668 | 713 | 0.17097 | 0.14527 |
| | | 5 | 2747 | 2912 | 0.17105 | 0.14803 |
| | | 6 | 10680 | 10690 | 0.17522 | 0.170083 |
| | | 7 | 38262 | 39022 | 0.17948 | 0.16997 |
| | | 8 | 79780 | 79735 | 0.18254 | 0.18014 |
| | | 9 | 97155 | 97154 | 0.18368 | 0.18306 |
| | | 10 | 99733 | 99733 | 0.18378 | 0.18372 |
| | | 11 | 99979 | 99979 | 0.18379 | 0.18379 |
| | | 12 | 99999 | 99999 | 0.18379 | 0.18379 |
| | | 14 | 100000 | 100000 | 0.18380 | 0.18380 |
| | | 16 | 100000 | 100000 | 0.18380 | 0.18380 |
| | 18 | 100000 | 100000 | 0.18380 | 0.18380 | |
| | 20 | 100000 | 100000 | 0.18380 | 0.18380 | |
| | 978 Actual | 0 | 5 | 5 | 0. | 0.282 |
| | | 1 | 10 | 8 | 0.17307 | 0.09999 |
| | | 2 | 17 | 17 | 0.15341 | 0.14836 |
| | | 3 | 34 | 34 | 0.14761 | 0.13990 |
| 4 | | 81 | 81 | 0.14891 | 0.14838 | |
| 5 | | 212 | 214 | 0.16455 | 0.14427 | |
| 6 | | 623 | 662 | 0.16454 | 0.13996 | |
| 7 | | 2019 | 2098 | 0.16230 | 0.13980 | |
| 8 | | 6896 | 7169 | 0.16506 | 0.14589 | |
| 9 | | 20611 | 20820 | 0.16747 | 0.16227 | |
| 10 | | 53243 | 53146 | 0.17133 | 0.16812 | |
| 11 | | 85830 | 85792 | 0.17387 | 0.17188 | |
| 12 | | 97629 | 97627 | 0.17452 | 0.17402 | |
| 14 | | 99976 | 99976 | 0.17464 | 0.17464 | |
| 16 | | 100000 | 100000 | 0.17464 | 0.17465 | |
| 18 | | 100000 | 100000 | 0.17464 | 0.17465 | |
| 20 | 100000 | 100000 | 0.17464 | 0.17465 | | |

5.2 二段階目への最大重みクリーク探索適用の評価

従来の整数計画法で集める候補頂点集合に対し最大重みクリーク探索を行う手法を、従来手法 [15] と比較する。一段階目からの入力クリークにはどち

らも最大重みクリーク探索による解を用いた。二段階目の各手法の計算時間は最大 18hr [15] とした。その他のパラメータも HMCAPIP 法 [15] と同様である。テスト構成数 (No.tests) とテスト情報量の標準偏差の θ_k における平均値 (Avg. SD_{TI}) を表 3 に示す。同程度のテスト構成数で、情報量の標準偏差を小さく抑えることができた。ここで、 $OC = 6$ 以下では、実験の計算時間内でテスト構成数の増加が収束していた。

表 3 二段階目のクリーク探索法の比較

| Test Length | Item Bank Size | No.tests | | | Avg. SD_{TI} | |
|-------------|----------------|---------------|----------------|----------------|----------------|-----------------|
| | | OC | HMCAPIP | Proposal | HMCAPIP | Proposal |
| 1000 | 1000 | 4 | 425 | 418 | 0.11397 | 0.092704 |
| | | 6 | 3137 | 3135 | 0.10744 | 0.09627 |
| | | 8 | 24127 | 24396 | 0.12254 | 0.08484 |
| | | 10 | 60022 | 60822 | 0.14784 | 0.11807 |
| | | 12 | 96331 | 99131 | 0.16425 | 0.15583 |
| | | 14 | 101040 | 104913 | 0.16644 | 0.15825 |
| | | 16 | 100398 | 101898 | 0.16677 | 0.16323 |
| | | 18 | 101862 | 101745 | 0.16687 | 0.16628 |
| | 20 | 102900 | 100400 | 0.16620 | 0.16672 | |
| | 978 Actual | 4 | 582 | 581 | 0.15680 | 0.11540 |
| | | 6 | 5031 | 5011 | 0.14336 | 0.12091 |
| | | 8 | 35519 | 35335 | 0.14174 | 0.10407 |
| | | 10 | 62227 | 62081 | 0.17645 | 0.17173 |
| | | 12 | 102529 | 100529 | 0.17126 | 0.17028 |
| 14 | | 100786 | 100844 | 0.17457 | 0.17431 | |
| 16 | | 107300 | 101900 | 0.17203 | 0.17503 | |
| 18 | | 101913 | 101900 | 0.17411 | 0.17338 | |
| 20 | 103100 | 100400 | 0.17276 | 0.17474 | | |

5.3 混合整数計画問題を用いる手法の評価

r のチューニング結果について表 4, 5, 6 に示す。結果より、 $OC = 8$ 以上では、離散化した r において、構成数の最大化と情報量のちらばりの最小化の完全最適解が存在し、その r はどれも 0.1 であった。 r を小さくすることで解の更なる改善が見込まれる。一方で、 $OC = 6$ 以下では、離散化した r において、構成数の最大化と情報量のちらばりの最小化に対するパレート解集合が存在した。比較的 OC 条件の厳しい $OC = 4$ の場合においては、パレート解は $r = 0.7, 0.8, 0.9$ 付近に存在し、比較的 OC 条件の厳しい $OC = 6$ の場合においては、パレート解

は $r = 0.3, 0.4, 0.5$ 付近に存在していた。したがって、所望の OC 条件が厳しいほど、決定変数数を大きく (r を大きく) する必要がある。

表 4 r によるテスト構成数と等質性の変化 ($r = 0.1 \sim 0.3$)

| Test Length | Item Bank | OC | Proposal | | | | | | |
|-------------|------------|------------|-----------|----------------|-----------|----------------|-----------|----------------|---------|
| | | | $r = 0.1$ | | $r = 0.2$ | | $r = 0.3$ | | |
| | | | No.tests | Avg. SD_{TI} | No.tests | Avg. SD_{TI} | No.tests | Avg. SD_{TI} | |
| 1000 | 1000 | 4 | 314 | 0.08614 | 353 | 0.04551 | 378 | 0.03220 | |
| | | 6 | 2340 | 0.07438 | 2775 | 0.03599 | 2963 | 0.02962 | |
| | | 8 | 26953 | 0.00533 | 25350 | 0.00652 | 20871 | 0.01249 | |
| | | 10 | 163927 | 0.00078 | 72192 | 0.04165 | 51069 | 0.09418 | |
| | | 12 | 201807 | 0.00919 | 123556 | 0.11090 | 95881 | 0.15939 | |
| | | 14 | 215401 | 0.00782 | 161792 | 0.07088 | 113209 | 0.13829 | |
| | | 16 | 214398 | 0.00933 | 183496 | 0.03782 | 124930 | 0.12141 | |
| | | 18 | 197366 | 0.02812 | 206766 | 0.00040 | 126983 | 0.11821 | |
| | 20 | 208100 | 0.01613 | 190000 | 0.02508 | 155900 | 0.07997 | | |
| | 978 Actual | 978 Actual | 4 | 419 | 0.08924 | 497 | 0.04450 | 525 | 0.02990 |
| | | | 6 | 3681 | 0.08172 | 4217 | 0.04276 | 4380 | 0.03676 |
| | | | 8 | 48695 | 0.00216 | 36417 | 0.00496 | 28682 | 0.00963 |
| | | | 10 | 195235 | 0.00083 | 88176 | 0.06082 | 70430 | 0.10555 |
| | | | 12 | 208520 | 0.00197 | 127009 | 0.12089 | 103116 | 0.16257 |
| | | | 14 | 224596 | 0.00070 | 166180 | 0.06942 | 113358 | 0.14815 |
| | | | 16 | 216800 | 0.00143 | 184599 | 0.04086 | 125231 | 0.13003 |
| 18 | | | 224422 | 0.00069 | 206019 | 0.00090 | 129523 | 0.12256 | |
| 20 | 215600 | 0.00119 | 191600 | 0.02378 | 155998 | 0.08530 | | | |

表 5 r によるテスト構成数と等質性の変化 ($r = 0.4 \sim 0.6$)

| Test Length | Item Bank | OC | Proposal | | | | | | |
|-------------|------------|------------|-----------|----------------|-----------|----------------|-----------|----------------|---------|
| | | | $r = 0.4$ | | $r = 0.5$ | | $r = 0.6$ | | |
| | | | No.tests | Avg. SD_{TI} | No.tests | Avg. SD_{TI} | No.tests | Avg. SD_{TI} | |
| 1000 | 1000 | 4 | 394 | 0.02738 | 406 | 0.02094 | 416 | 0.02047 | |
| | | 6 | 3056 | 0.03003 | 3063 | 0.03131 | 3011 | 0.03395 | |
| | | 8 | 16208 | 0.01545 | 20029 | 0.01447 | 12914 | 0.02595 | |
| | | 10 | 46533 | 0.10864 | 43845 | 0.11791 | 42590 | 0.12267 | |
| | | 12 | 97391 | 0.15532 | 100239 | 0.14868 | 95888 | 0.15922 | |
| | | 14 | 104986 | 0.15408 | 102746 | 0.15914 | 101927 | 0.16111 | |
| | | 16 | 107587 | 0.14927 | 112677 | 0.14018 | 102500 | 0.16035 | |
| | | 18 | 109199 | 0.14794 | 105335 | 0.15574 | 103738 | 0.15941 | |
| | 20 | 117821 | 0.13264 | 118349 | 0.13264 | 104084 | 0.13208 | | |
| | 978 Actual | 978 Actual | 4 | 544 | 0.02157 | 562 | 0.01926 | 577 | 0.02301 |
| | | | 6 | 4488 | 0.03720 | 4440 | 0.03988 | 4491 | 0.04251 |
| | | | 8 | 17574 | 0.02737 | 16833 | 0.03881 | 13397 | 0.09316 |
| | | | 10 | 61118 | 0.13193 | 59175 | 0.13785 | 61037 | 0.13197 |
| | | | 12 | 101203 | 0.16637 | 101133 | 0.16647 | 99261 | 0.17033 |
| | | | 14 | 106882 | 0.16016 | 105079 | 0.16371 | 102116 | 0.1702 |
| | | | 16 | 107991 | 0.16062 | 107682 | 0.16108 | 103185 | 0.16981 |
| 18 | | | 113221 | 0.14921 | 106451 | 0.16173 | 102973 | 0.16906 | |
| 20 | 118619 | 0.14005 | 110269 | 0.15496 | 104852 | 0.16478 | | | |

表 6 r によるテスト構成数と等質性の変化 ($r = 0.7 \sim 0.9$)

| Test Length | Item Bank | OC | Proposal | | | | | | |
|-------------|------------|------------|-----------|----------------|-----------|----------------|-----------|----------------|---------|
| | | | $r = 0.7$ | | $r = 0.8$ | | $r = 0.9$ | | |
| | | | No.tests | Avg. SD_{TI} | No.tests | Avg. SD_{TI} | No.tests | Avg. SD_{TI} | |
| 1000 | 1000 | 4 | 423 | 0.01716 | 425 | 0.01845 | 436 | 0.02310 | |
| | | 6 | 2896 | 0.03734 | 2842 | 0.03963 | 2840 | 0.04426 | |
| | | 8 | 8605 | 0.08741 | 8233 | 0.09889 | 9250 | 0.07467 | |
| | | 10 | 37619 | 0.14763 | 37382 | 0.14972 | 37337 | 0.15026 | |
| | | 12 | 95062 | 0.16139 | 94763 | 0.16231 | 94114 | 0.16461 | |
| | | 14 | 103309 | 0.15775 | 100598 | 0.16473 | 100128 | 0.16582 | |
| | | 16 | 102268 | 0.16072 | 101659 | 0.16227 | 100190 | 0.16672 | |
| | | 18 | 104221 | 0.15823 | 101560 | 0.16552 | 101472 | 0.16582 | |
| | 20 | 102889 | 0.15958 | 105032 | 0.15463 | 100348 | 0.16667 | | |
| | 978 Actual | 978 Actual | 4 | 586 | 0.02230 | 597 | 0.02226 | 601 | 0.02430 |
| | | | 6 | 4446 | 0.04606 | 4336 | 0.04866 | 4288 | 0.05004 |
| | | | 8 | 11250 | 0.12663 | 12839 | 0.09463 | 10724 | 0.13965 |
| | | | 10 | 55980 | 0.14829 | 55349 | 0.15046 | 54562 | 0.15316 |
| | | | 12 | 99608 | 0.16958 | 98541 | 0.17216 | 98035 | 0.17374 |
| | | | 14 | 100975 | 0.17310 | 100640 | 0.17413 | 100727 | 0.17384 |
| | | | 16 | 105306 | 0.16558 | 101905 | 0.17319 | 102832 | 0.17051 |
| 18 | | | 101505 | 0.17265 | 101125 | 0.17380 | 101725 | 0.17200 | |
| 20 | 105220 | 0.16402 | 102827 | 0.16865 | 105282 | 0.16308 | | | |

続いて、最適な r における提案手法と、従来手法の比較を表 7 に示す。OC = 4, 6 については、パレート解集合の中でテスト構成数が最大となる r を採用した。提案手法は、テスト構成数において、OC = 6 以外の全ての条件で従来手法を上回った。また、全ての条件において測定精度の散らばりを小さく抑えた。また、OC = 4, 6 のみ制限時間内に構成数が収束しており、終盤には測定精度が目標値から離れたテストも追加されている。しかし他条件と比べて等質化の効果は弱まっているものの、同程度の構成数と等質性の向上の 2 点を実現している。

表 7 混合整数計画問題用いる手法と従来手法の比較

| Test Length | Item Bank Size | No.tests | | | Avg. SD_{TI} | | |
|-------------|----------------|---------------|-------------|----------------|----------------|----------------|----------------|
| | | OC | HMCAPIP | Proposal | HMCAPIP | Proposal | |
| 1000 | 1000 | 4 | 425 | 436 | 0.11397 | 0.02310 | |
| | | 6 | 3137 | 3063 | 0.10744 | 0.03131 | |
| | | 8 | 24127 | 26953 | 0.12254 | 0.00533 | |
| | | 10 | 60022 | 163927 | 0.14784 | 0.00078 | |
| | | 12 | 96331 | 201807 | 0.16425 | 0.00919 | |
| | | 14 | 101040 | 215401 | 0.16644 | 0.00782 | |
| | | 16 | 100398 | 214398 | 0.16677 | 0.00933 | |
| | | 18 | 101862 | 206766 | 0.16687 | 0.0004 | |
| | | 20 | 102900 | 208100 | 0.1662 | 0.01613 | |
| | 978 Actual | 978 Actual | 4 | 582 | 601 | 0.15680 | 0.0243 |
| | | | 6 | 5031 | 4491 | 0.14336 | 0.04251 |
| | | | 8 | 35519 | 48695 | 0.14174 | 0.00216 |
| | | | 10 | 62227 | 195235 | 0.17645 | 0.00083 |
| | | | 12 | 102529 | 208520 | 0.17126 | 0.00197 |
| | | | 14 | 100786 | 224596 | 0.17457 | 0.00070 |
| | | | 16 | 107300 | 216800 | 0.17203 | 0.00143 |
| 18 | 101913 | 224422 | 0.17411 | 0.00069 | | | |
| 20 | 103100 | 215600 | 0.17276 | 0.00119 | | | |

次に、能力値パラメータごとのテスト情報量の標準偏差を表 8 に示す。提案手法は、全ての能力値パラメータにおいてテスト情報量の標準偏差を小さくすることができている。

表 8 各手法における能力値パラメータごとのテスト情報量の目標値からの差の絶対値の比較

| Item Bank Size | $\theta = -2.0$ | | $\theta = -1.0$ | | $\theta = 0.0$ | | $\theta = 1.0$ | | $\theta = 2.0$ | | |
|----------------------|-----------------|----------|-----------------|----------|----------------|----------|----------------|----------|-----------------|----------|----------------|
| | HMCAPIP | Proposal | HMCAPIP | Proposal | HMCAPIP | Proposal | HMCAPIP | Proposal | HMCAPIP | Proposal | |
| 1000 | oc | | | | | | | | | | |
| | 4 | 0.06675 | 0.02624 | 0.12270 | 0.02176 | 0.08143 | 0.02063 | 0.10496 | 0.02227 | 0.19402 | 0.02458 |
| | 6 | 0.06285 | 0.03228 | 0.1125 | 0.03053 | 0.06557 | 0.03195 | 0.09521 | 0.03044 | 0.20106 | 0.03137 |
| | 8 | 0.08454 | 0.00560 | 0.12599 | 0.00526 | 0.08614 | 0.00515 | 0.11033 | 0.00521 | 0.20569 | 0.00542 |
| | 10 | 0.12484 | 0.00077 | 0.15152 | 0.00078 | 0.12682 | 0.00077 | 0.14508 | 0.00078 | 0.19093 | 0.00078 |
| | 12 | 0.14277 | 0.01335 | 0.16841 | 0.00872 | 0.14159 | 0.00779 | 0.15921 | 0.00821 | 0.20929 | 0.00788 |
| | 14 | 0.14520 | 0.01345 | 0.17077 | 0.00664 | 0.14389 | 0.00596 | 0.16105 | 0.00656 | 0.21131 | 0.00650 |
| | 16 | 0.14181 | 0.01363 | 0.16647 | 0.00862 | 0.13811 | 0.00803 | 0.15773 | 0.00830 | 0.21201 | 0.00805 |
| | 18 | 0.14525 | 0.03394 | 0.17096 | 0.02751 | 0.14391 | 0.02195 | 0.16125 | 0.02462 | 0.21296 | 0.03260 |
| | 20 | 0.14512 | 0.02160 | 0.17089 | 0.01490 | 0.14372 | 0.01436 | 0.16116 | 0.01491 | 0.21269 | 0.01487 |
| 978 | 4 | 0.19954 | 0.02537 | 0.16778 | 0.02068 | 0.10986 | 0.02219 | 0.09062 | 0.02492 | 0.21618 | 0.02832 |
| | 6 | 0.18114 | 0.03939 | 0.15046 | 0.03392 | 0.10651 | 0.03494 | 0.08771 | 0.03415 | 0.19100 | 0.04361 |
| | 8 | 0.16098 | 0.00230 | 0.13465 | 0.00214 | 0.12986 | 0.00202 | 0.11604 | 0.002070 | 0.16718 | 0.00228 |
| | 10 | 0.23335 | 0.00093 | 0.18425 | 0.00081 | 0.13451 | 0.00075 | 0.09627 | 0.00077 | 0.23385 | 0.00091 |
| | 12 | 0.19625 | 0.00114 | 0.17635 | 0.00144 | 0.14935 | 0.00236 | 0.14149 | 0.003440 | 0.19285 | 0.00148 |
| | 14 | 0.20073 | 0.00075 | 0.18134 | 0.00069 | 0.15160 | 0.00065 | 0.14276 | 0.000670 | 0.19643 | 0.00074 |
| Actual | 16 | 0.19704 | 0.00091 | 0.17706 | 0.00110 | 0.15004 | 0.00169 | 0.14226 | 0.00238 | 0.19377 | 0.00106 |
| | 18 | 0.20004 | 0.00075 | 0.18062 | 0.00068 | 0.15138 | 0.00064 | 0.14262 | 0.00066 | 0.19591 | 0.00074 |
| | 20 | 0.19813 | 0.00087 | 0.17842 | 0.00096 | 0.15044 | 0.00135 | 0.14235 | 0.00178 | 0.19447 | 0.00097 |

5.4 RndMIP と RndIP の混合探索手法の評価

$OC = 6$ のとき, r による決定変数数のチューニングのみでは従来手法よりもテスト構成数を増やすことができない. そこで, 二段階目の候補頂点集合の探索において, 決定変数を減らして行う従来の整数計画問題と提案の混合整数計画問題を混合する. 混合割合 $retio_{MIP}$ のチューニング結果について表 9 に示す. また, 最適な $retio_{MIP}$ における提案手法と, 従来手法の比較を表 10 に示す. 提案手法 (RMRIM 法) によって, 従来手法と同様のテスト構成数で, 測定精度のちらばりを抑えることができた.

表 9 $retio_{MIP}$ によるテスト構成数と等質性の変化

| Item Bank | OC | r | Proposal (RMRIM 法) | | | | | | | |
|------------|----|-----|---------------------|----------------|---------------------|----------------|---------------------|----------------|---------------------|----------------|
| | | | $retio_{MIP} = 0.2$ | | $retio_{MIP} = 0.4$ | | $retio_{MIP} = 0.6$ | | $retio_{MIP} = 0.8$ | |
| | | | No.tests | Avg. SD_{TI} | No.tests | Avg. SD_{TI} | No.tests | Avg. SD_{TI} | No.tests | Avg. SD_{TI} |
| 1000 | 6 | 0.3 | 2968 | 0.09072 | 3002 | 0.08119 | 3004 | 0.06352 | 3035 | 0.03190 |
| | | 0.4 | 3058 | 0.08979 | 3079 | 0.08286 | 3077 | 0.06818 | 3134 | 0.03531 |
| | | 0.5 | 3109 | 0.09078 | 3104 | 0.08428 | 3118 | 0.06891 | 3132 | 0.04492 |
| 978 Actual | 6 | 0.3 | 4863 | 0.11467 | 4974 | 0.11159 | 4880 | 0.10085 | 4773 | 0.08244 |
| | | 0.4 | 4997 | 0.11457 | 4854 | 0.10886 | 4804 | 0.09561 | 4646 | 0.10690 |
| | | 0.6 | 5055 | 0.11626 | 4790 | 0.10699 | 4697 | 0.09462 | 4584 | 0.07676 |

表 10 混合手法と従来手法の比較

| Item Bank Size | OC | No.tests | | | Avg. SD_{TI} | | |
|----------------|----|-------------|----------|-------------|----------------|----------------|---------|
| | | HMCAPIP | RndMIP 法 | RMRIM 法 | HMCAPIP | RndMIP 法 | RMRIM 法 |
| 1000 | 6 | 3137 | 3063 | 3134 | 0.10744 | 0.03131 | 0.03531 |
| 978 | 6 | 5031 | 4491 | 5055 | 0.14336 | 0.04251 | 0.11626 |

6 むすび

本論文では e-Testing における自動テスト構成のための最大重みクリーク探索と混合整数計画問題を用いる手法を提案した. 具体的には, 頂点をテスト, 辺をテスト間重複項目数制約の適否, 頂点の重みを目標とする測定精度からの差異分のペナルティとしたグラフから最大重みクリーク探索を行う. そして, 二段階目のクリーク追加の候補頂点集合の探索においても, 目標測定精度からの差異を最

小化する混合整数計画問題を解く．ここで，アイテムバンクからランダムに抽出する部分集合からの探索を行うことで，問題の解空間の次元を削減した．その結果，提案手法はテスト構成数と測定精度の等質性の両方を従来手法よりも向上させることができた．また，近年，各受検者に対し測定精度を減少させずに出題項目数を減少することができる技術として適応型テストが知られている [7], [29]．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，適応型テストでは各受検者の等質性が保証されていない点が問題となっている．この問題を解決する手法として，HMCAPIP 法 [15] を用いた，二段階等質適応型テストが提案されている [29]．この適応型テストに，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各テストの測定精度をより等質にし，テストの信頼性を保証することができる．

参考文献

- [1] Maomi Ueno. Ai based e-testing as a common yardstick for measuring human abilities. In *2021 18th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 1–5. IEEE, 2021.
- [2] Maomi Ueno, Kazuma Fuchimoto, and Emiko Tsutsumi. E-testing from artificial intelligence approach. *Behaviormetrika*, 48(2):409–424, 2021.
- [3] 仁田 善雄, 齋藤 宣彦, 後藤 英司, 高木 康, 石田 達樹, and 江藤 一洋. 医療系大学間共用試験における e テスティング. In **日本テスト学会 第 12 回大会 発表論文抄録集**, pages 58–59, 2014.
- [4] 谷澤 明紀 and 本多 康弘. 情報処理技術者試験における e テスティング. In **日本テスト学会 第 12 回大会 発表論文抄録集**, pages 54–57, 2014.
- [5] Frederic M Lord and Melvin R Novick. *Statistical theories of mental test scores*. IAP, 2008.
- [6] Fumiko Samejima. Weakly parallel tests in latent trait theory with some criticisms of classical test theory. *Psychometrika*, 42(2):193–198, 1977.

- [7] Wim J Van der Linden. *Handbook of item response theory: Three volume set*. CRC Press, 2018.
- [8] Ellen Boekkooi-Timminga. The construction of parallel tests from irt-based item banks. *Journal of Educational Statistics*, 15(2):129–145, 1990.
- [9] Ronald D Armstrong, Douglas H Jones, and Zhaobo Wang. Automated parallel test construction using classical test theory. *Journal of Educational Statistics*, 19(1):73–90, 1994.
- [10] Ronald D Armstrong, Douglas H Jones, and Charles S Kuncce. Irt test assembly using network-flow programming. *Applied Psychological Measurement*, 22(3):237–247, 1998.
- [11] Wim J van der Linden and Jos J Adema. Simultaneous assembly of multiple test forms. *Journal of educational measurement*, 35(3):185–198, 1998.
- [12] Wim J van der Linden. *Linear Models for Optimal Test Design*. Springer, 2005.
- [13] Pokpong Songmuang and Maomi Ueno. Bees algorithm for construction of multiple test forms in e-testing. *IEEE Transactions on Learning Technologies*, 4(3):209–221, 2010.
- [14] Takatoshi Ishii, Pokpong Songmuang, and Maomi Ueno. Maximum clique algorithm and its approximation for uniform test form assembly. *IEEE Transactions on Learning Technologies*, 7(1):83–95, 2014.
- [15] Kazuma Fuchimoto, Takatoshi Ishii, and Maomi Ueno. Hybrid maximum clique algorithm using parallel integer programming for uniform test assembly. *IEEE Transactions on Learning Technologies*, 15(2):252–264, 2022.
- [16] Howard Wainer. Cats: Whither and whence. *Psicologica*, 21(1):121–133, 2000.
- [17] Takatoshi Ishii, Pokpong Songmuang, and Maomi Ueno. Maximum clique algorithm for uniform test forms assembly. In *International Conference*

- on Artificial Intelligence in Education*, pages 451–462. Springer, 2013.
- [18] Takatoshi Ishii and Maomi Ueno. Algorithm for uniform test assembly using a maximum clique problem and integer programming. In *International Conference on Artificial Intelligence in Education*, pages 102–112. Springer, 2017.
- [19] 淵本 壱真 and 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*, 103(12):881–893, 2020.
- [20] Etsuji Tomita, Sora Matsuzaki, Atsuki Nagao, Hiro Ito, and Mitsuo Wakatsuki. A much faster algorithm for finding a maximum clique with computational experiments. *Journal of Information Processing*, 25:667–677, 2017.
- [21] Chu Min Li, Hua Jiang, and Felip Manyà. On minimization of the number of branches in branch-and-bound algorithms for the maximum clique problem. *Computers & Operations Research*, 84:1–15, 2017.
- [22] 石井 隆稔, 赤倉 貴子, and 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*, 100(1):47–59, 2017.
- [23] Frank B Baker and Seock Ho Kim. *Item response theory: Parameter estimation techniques*. CRC Press, 2004.
- [24] Dmitry I Belov and Ronald D Armstrong. A constraint programming approach to extract the maximum number of non-overlapping test forms. *Computational Optimization and Applications*, 33(2-3):319–332, 2006.
- [25] 植野 真臣 and 永岡 慶三. *e テスティング*. 培風館, 2009.
- [26] Dmitry I Belov. Uniform test assembly. *Psychometrika*, 73(1):21, 2008.
- [27] Shaowei Cai and Jinkun Lin. Fast solving maximum weight clique problem in massive graphs. In *IJCAI*, pages 568–574, 2016.
- [28] Pablo San Segundo, Fabio Furini, and Jorge Artieda. A new branch-and-bound algorithm for the maximum weighted clique problem. *Computers & Operations Research*, 110:18–33, 2019.

- [29] Wakaba Kishida, Kazuma Fuchimoto, Yoshimitsu Miyazawa, and Maomi Ueno. Item difficulty constrained uniform adaptive testing. In *International Conference on Artificial Intelligence in Education*, pages 568–573. Springer, 2023.