1C4-1

未知の選好を含む最大安定度マッチング問題の定式化
Formalization of Almost Stable Matching Problem Including Unknown Preferences

境　良太　　松原　繁夫
Ryota Sakai　　Shigeo Matsubara

京都大学情報学研究科社会情報学専攻
Department of Social Informatics, Kyoto University

A matching problem of multiple tasks and contractors can be formulated as a stable marriage problem. In large scale matching problems, it becomes difficult for tasks and contractors to tell their own preferences because evaluating all the contractors/tasks are costly for tasks/contractors. These unknown preferences may cause mis-matching and make the matching unstable. To solve the stable marriage problem with unknown preferences, we expand stable marriage problems to that including unknown preferences. In this problem, we are difficult to know whether a matching is stable or unstable because we cannot know if mis-matching exists. We define almost stable matching as the matching having the least value in the expected numbers of mis-matching. We propose an algorithm to find an almost stable matching.

1. はじめに

人に対するタスクの割り当て問題において，誰にどのタスクを割り当てるかということは大きな問題となる．なぜならば，タスク依頼者はそれぞれのタスクに必要な能力を持つタスク請負人にタスクを割り当てて欲しい一方で，それぞれの請負人は，得意なタスクや好むタスクを持つなどの選好を持っており，両者の要望を満たす必要があるためである．

この問題を，一人に一つのタスクを割り当てる問題と考えると，安定結婚問題[Gale 1962]として広く知られたマッチングの問題に帰着して解くことができる．もしマッチングにミスマッチが起こると，ミスマッチの起こったペアで結託して新しいペアを作ってしまい，不安定なマッチングとなる．これは，タスクや請負人の要望が満たされていないため起こる．安定結婚問題とは，ミスマッチの起こらない安定マッチングを求める問題である．

しかし，安定結婚問題として解くためには，全員の全員に対する選好が必要であるが，一般に，全員に対する選好を得ることは難しい．特に大規模なタスク割り当て問題においては，請負人がすべてのタスクを精査すると非常にコストがかかるため，全員に対する正確な選好を得ることは現実的でない．

このような状況で選好を得る方法としては，タスクをカテゴリに分けて，カテゴリを精査することでカテゴリ間の選好順序を決めることが考えられる．すると，これは同順位を許す安定結婚問題[Irving 1994]として考えることができる．しかし，カテゴリ分けを行っても，カテゴリ内のタスク同士の選好順序は未知である．選好が未知であると，ミスマッチの存在も未知となり，さらにはマッチングの安定性も未知となってしまう．未知のミスマッチそれ自体は問題を起こさないかもしれないが，時間の経過によって選好の知識を得ることで，それまで未知であったミスマッチに気づき，マッチング結果に不満が生ずる．既存の方法では，必ずしもこれに十分対処できるマッチングを得ることはできない．

これらの問題を解決するために，未知の選好の定式化を行った．未知の選好を確率的に扱い，確率的に最もミスマッチの少ないマッチングを最大安定度マッチングと定義した．そして，未知の選好を含む安定結婚問題に対し最大安定度マッチングを求めるアルゴリズムを作成した．

連絡先：〒606-8501 京都府京都市左京区吉田本町京都大学大学院情報学研究科社会情報学専攻 石田・松原研究室

2. 関連研究

2.1 安定結婚問題

安定結婚問題とは，男性と女性，タスクと請負人のような二つの異なる集団を 1 対 1 マッチングさせる問題である．それぞれの集団のそれぞれのプレーヤーは，相手の集団のそれぞれのプレーヤーに対する選好順序を持っており，それぞれのプレーヤーの選好を考慮してマッチングをしなければならない．

あるマッチングが成立した時，そのマッチングに含まれないペアで，現在のマッチングのパートナーよりお互いを好むものをブロッキングペアという．ブロッキングペアが存在すると両者の選好が満たされず，マッチングが不安定となってしまう．逆に，ブロッキングペアが存在しないマッチングが安定マッチングであると定義されている．

選好順序が与えられた時に安定マッチングを求める問題を安定結婚問題と呼ぶ．Gale，Shapley[Gale 1962]によって，それぞれのプレーヤーがどのような選好順序を持っていたとしても必ず安定マッチングが存在することが証明されている．またその証明の過程から，安定マッチングを求めるアルゴリズムも知られている．

2.2 安定結婚問題の拡張

安定結婚問題の拡張として，同順位を許す安定結婚問題[Irving 1994]や，パートナーになることを拒否することを許す安定結婚問題が知られている．本研究では，提案するアルゴリズムにおいてこれらの拡張を用いるため説明をする，

まず，同順位を許す安定結婚問題について述べる．元の安定結婚問題では，どの二人のプレーヤーに対しても選好順序が存在したが，同順位を許す安定結婚問題では，二人のプレーヤーを同程度に選好するということが許される．選好に同順位が許されることにより，ブロッキングペアの定義も見直されている．同順位を考慮すると，例えば，マッチングに含まれないペアで，現在のマッチングのパートナーより，同等かそれ以上にペアの相手を好むものを，ブロッキングペアとすることができる．この時の安定マッチングは超安定マッチングと呼ばれている．この他の種類の安定性も存在するが，本論文では扱わない．元の安定結婚問題では必ず安定マッチングが存在するが，同順位

を許す安定結婚問題において，超安定マッチングは必ずしも存在しない．超安定マッチングの存在を判定し，存在すればそのマッチングを返す，O(n^2)のアルゴリズムが知られている [Irving 1994].

次に， パートナーになることを拒否することを許す安定結婚問題について述べる．パートナーになることを拒否することを許す場合，ある人とペアとなるよりも，一人でいる方が望ましいという状態を表わすことができる．パートナーになることを拒否することを許す問題の特徴としてあげられるのは，必ずしも全員がマッチングされるとは限らないことである．これは，もし誰ともパートナーになりたくない人がいれば，その人を含むマッチングが存在しないことなどから理解できる．

2.3 マッチングの不安定度

元の安定結婚問題では，必ず安定マッチングが存在するが，同順位を許す安定結婚問題の超安定マッチングのように，問題の拡張によっては安定マッチングが存在しない場合がある．未知の選好を含む場合にも，後に述べるように必ずしも安定マッチングは存在しない．しかし，安定マッチングが存在しない場合であっても，実際にはより安定となるマッチングが必要とされている．そのため，不安定の度合いを測る尺度が必要となる．

不安定度の尺度の候補としては，ブロッキングペアの数や，ブロッキングペアに含まれる男女の人数などが考えられる．しかし，ミスマッチの気づきやすさという観点から，ブロッキングペアの数を尺度にするのがよいとされている[Eriksson 2008]．そのため，本論文では，ブロッキングペアの数が少ないほどマッチングが安定であるとみなす．

2.4 安定ルームメイト問題

二人部屋の寮における部屋の割り当ての問題のように，ひとつのグループ内のメンバー同士を選好に基づいてマッチングさせる安定ルームメイト問題も，Gale，Shapley によって提案されている．安定ルームメイト問題においては，必ずしも安定マッチングは存在しない[Gale 1962].

そこで，Abraham はブロッキングペアの数が K であるようなマッチングを求めるアルゴリズムを提案した[Abraham 2006]．このアルゴリズムでは，K が定数として与えられたとき，ブロッキングペアの数が K であるマッチングを多項式時間で求めることができる．

Abraham のアルゴリズムでは，まず，ブロッキングペアの数が K となるようブロッキングペアを予め定めておき，そのペアがブロッキングペアとなるのを阻むペア以外のペアのみを用いて安定マッチングを求めることで，予め定めたマッチングをブロッキングペアとするマッチングを求めている．もし，どのようにブロッキングペアを定めても安定マッチングが求まらない場合は，ブロッキングペアの数が K のマッチングは存在しない．

3. 未知の選好を含むマッチング

3.1 未知の選好の定式化

タスク割り当て問題におけるタスクと請負人のマッチングの例をもとに，未知の選好を含む場合に最大安定度マッチングを求める問題の定式化を提案する．

元の安定結婚問題と同様，n 個のタスクと，n 人の請負人が存在するときに，タスクと請負人に対して 1 対 1 マッチングを行う．そして，安定マッチングを得ることを目標とする．しかし，相手に対する知識を得て選好順序を決定することはコストがかかるため，必ずしも他のタスクや請負人全員に対する選好順序は得られないとする．そのため，マッチングを解くためには，その選好順序を未知のものとして扱わなければならない．なお，選好が未知となる原因は知識の不足であるため，相手に対する知識を十分得ることができれば，選好順序を述べることはできる．そのため，未知の選好順序に対しても，実際の選好順序は存在する．

t_1:(c_1 c_2)c_3	c_1:(t_1 t_2 t_3)
t_2: c_1(c_3 c_2)	c_2: t_1 t_3 t_2
t_3:(c_2 c_1)c_3	c_3: t_3 t_2 t_1

図 1: 未知の選好の例

t_1: c_1 c_2	c_1:(t_1 t_2)
t_2: c_1 c_2	c_2: t_1 t_2

図 2： 必ず安定となるマッチングの存在しない問題例

未知の選好を定式化するにあたって，簡単のため次の二つを仮定する．

- 各タスク，請負人が実際に持つ選好順序は，全員に対する選好順で，同順位を含まない．
- ある請負人の持つ，タスク t_1 と t_2 に対する選好順序が未知で，かつ，タスク t_2 と t_3 に対する選好順序が未知であれば，タスク t_1 と t_3 の選好順序も未知であるという推移律が成り立つ．

二つ目の仮定により，未知の選好は，図 1 のような，同順位を許す安定結婚問題と同様の形式で書くことができる．これは，タスクや請負人をカテゴライズして，カテゴリ同士の選好順序は得られるが，カテゴリ内の個々の相手同士の選好順序は未知のままであるという状況を，表現することができる．

3.2 未知の選好の導入による課題

この問題において課題となるのは，未知であった選好に対しても，実際の選好順序が存在するということである．もし未知の選好を無視してマッチングを行うと，未知の選好持つ実際の選好によりミスマッチが起こる可能性がある．選好が未知である間はそのミスマッチも未知であるが，時間の経過などによって，タスクの依頼者や請負人が選好の知識を得ると，ミスマッチに気づき，マッチングに不満が生じてしまう．そのため，未知の選好に対して，考えられる実際の選好順序によらず，ミスマッチが起こらないことが求められている．

しかし，実際の選好によらず必ず安定であるようなマッチングが求められている一方で，そのようなマッチングは必ずしも存在しないことが判明した．例えば，図 2 の問題例において，(t_1, c_1) (t_2, c_2)のマッチングを成立させた時，もし，c_1 が t_2 より t_1 を好めば安定となるが，もし，t_1 より t_2 を好めば(t_2, c_1)がブロッキングペアとなり，不安定なマッチングとなってしまう．そのため，このマッチングは不安定となる可能性がある．一方，(t_1, c_2) (t_2, c_1)のマッチングも同様に考えることができ，不安定となる可能性がある．そのため，この問題例では，必ず安定であるようなマッチングは存在しない．

必ず安定であるようなマッチングが必ずしも得られるとは限らないため，マッチングの不安定さの度合いをもとに，できるだけ安定となるマッチングを決定することを考えなければならない．マッチングの不安定度として，ブロッキングペアの数を用いればよいことは述べたが，この問題では，ブロッキングペアの有無も未知となってしまう．

そこで，未知の選好を確率的に扱うことを考えた．未知の選好を確率的に扱うことで，ブロッキングペアの数の期待値を不安

定度とするができる. 本研究では, ブロッキングペアの数の期待値を最小化するマッチングを最大安定度マッチングとした.

3.3 ブロッキングペアの数の期待値

考えられる様々なマッチングを, 不安定度の観点で比較するために, ブロッキングペアの数の期待値を定義する. 例えば, マッチング M_1 と M_2 を比較し, M_1 の方がブロッキングペアの期待値が小さければ, マッチング M_1 の方がより安定で望ましいマッチングとなる.

ブロッキングペアの数の期待値を求めるために, まず, あるマッチング M において, 任意のペア p がマッチング M のもとでブロッキングペアである確率 $P_M(p)$を定義する. この $P_M(p)$は, 例えば $P_M(p)$が 1/2 であれば, ペア p はマッチング M において 1/2 の確率でブロッキングペアになるということである.

ペア p を, マッチング M のパートナーとペア p の相手の間の選好順序について場合分けをし, それぞれのペア p について $P_M(p)$の値を求める.

- ペア p のタスク, 請負人のどちらかが, ペア p の相手よりマッチング M のパートナーを選好することが既知である場合, p はブロッキングペアとならず, $P_M(p) = 0$ となる. この集合を B_0 と定義する.
- ペア p のタスク, 請負人の双方が, マッチング M のパートナーよりペア p の相手を選好することが既知である場合, ペア p は互いをマッチングのパートナーより好むため p はブロッキングペアとなる. そのため, $P_M(p) = 1$ となる. このようなペアの集合を B_1 と定義する.
- ペア p のタスクはマッチング M のパートナーよりペア p の相手を選好するが, 請負人はマッチング M のパートナーとペア p の相手のどちらを選好するか未知である場合, 請負人はどちらのタスクを好むかが未知である. そこで, パートナーを選好する確率と, ペア p の相手を選好する確率をそれぞれ 1/2 と考えると, $P_M(p) =1/2$ となる. このようなペアの集合を B_2 と定義する.
- ペア p のタスクはマッチング M のパートナーとペア p の相手のどちらを選好するか未知であるが, 請負人はマッチング M のパートナーのタスクよりペア p の相手の請負人を選好する場合, B_2 の場合と同様に $P_M(p) =1/2$ となる. このようなペアの集合を B_3 と定義する.
- ペア p のタスク, 請負人の双方が, マッチング M のパートナーとペア p の相手の選好順序について未知である場合, タスク, 請負人のそれぞれ, ペアの相手を好む確率が 1/2 で, 両者が互いに選好してブロッキングペアとなる確率は 1/4 となり, $P_M(p) =1/4$ となる. このようなペアの集合を B_4 と定義する.

どのペアも上記の 5 通りのどれかに該当するため, すべてのペアに対し, $P_M(p)$を求めることができた.

$P_M(p)$を用いて, 今度はブロッキングペアの数の期待値を求める. それぞれのペアは $P_M(p)$の確率でブロッキングペアとなるため, すべてのペアに対する $P_M(p)$を足し合わせた値が, すべてのペアにおけるブロッキングペアの数の期待値となる. すなわち, マッチング M のブロッキングペアの数の期待値となっている. これを E_M と定義する.

$E_M = 0$ の場合, すべてのペアに対して $P_M(p) = 0$となり, ブロッキングペアが存在しないため, 必ず安定となる. 一方で, $E_M > 0$ ならば, 最低一つのペアがブロッキングペアとなる可能性があり, そのマッチングは安定とならない場合がある.

まとめると, 安定結婚問題に未知の選好を導入し, 同順位を許す安定結婚問題と同様の形式で表わす. しかし, 実際には選好が未知であっても, 本当の選好を持つため, 安定マッチングを持たない可能性がある. そこで, ブロッキングペアの数の期待値 E_M を導入し, E_M を最小化する最大安定度マッチングを求めることを問題の目的とした.

4. アルゴリズム

4 章では, ブロッキングペアの期待値 E_M を最小化するマッチング M を求めるアルゴリズムについて説明する. このアルゴリズムは, 安定ルームメイト問題における Abraham のアルゴリズム [Abraham 2006]を, 未知の選好を含む安定結婚問題に適用できるよう改良したものである. 本研究で行った主な工夫は, ブロッキングペアの数でなく, その期待値を扱えるようにしたこと, 及び, 未知の選好を扱えるよう, ペアの削除の条件を求めたことである.

まず, 3.3 節で述べたように, どのペア p に対しても, $P_M(p)$は 0, 1/2, 1/4, 1 はいずれかの値をとることが判明したが, 期待値を扱うために, これを用いる. $4E_M$ は整数となるため, Abraham のアルゴリズムと同様に, $4E_M = K$ となるようにブロッキングペアを予め定めておき, そのペアがブロッキングペアとなるようにマッチングを構成するという方針で解くことができる. もし, 予め定めたペアがブロッキングペアなるマッチングが存在すれば, そのマッチングはブロッキングペアの期待値 $E_M = K/4$ となる. 一方で, どのようにブロッキングペアを予め定めておいても, そのペアをブロッキングペアとするマッチングを作ることができなければ, $E_M = K/4$ となるマッチングは存在しない. これを $K = 0, 1, \ldots$と増やしながら実行していくことで, ブロッキングペアの数の期待値が最小のマッチングを求めることができる.

では, 図 3 に示した, 具体的なアルゴリズム MATCHING に関して説明をする. はじめに, 整数 K が与えられる. これをもとに, タスクと請負人のすべてペアを B_0～B_4 に振り分ける. ただし, 期待値 $E_M= K/4$ となるようにしたいため,

$$E_M = |B_1| + |B_2| / 2 + |B_3| / 2 + |B_4| / 4 = K / 4$$

を満たすように振り分けを行う. $B:=B_1 \cup B_2 \cup B_3 \cup B_4$ はブロッキングペアの集合となり, B をブロッキングペアとするようなマッチングを構成していく. そのために, すべての組み合わせのペアの集合を Q とし, B がブロッキングペアとなるのを阻むペアを Q から削除し, Q の残りのペアで安定マッチングを求めることを考える.

まず, あるペア$(t, c) \in B$ がマッチング M に含まれていると, M においてに(t, c)はブロッキングペアではなくなってしまう. そのため, (t, c)をペアの集合 Q から削除する. 次に, ペア(t, c)が B_1 や B_2 に含まれる時, ペア(t, c)がブロッキングペアであるためには, タスク t はマッチング M でのパートナーより請負人 c を好むことが必要である. そのため, タスク t が請負人 c より選好する請負人を c_1 とすると, マッチング M にはペア(t, c_1)が存在してはいけない. そのため, Q からペア(t, c_1)を削除する.

ペア (t, c_1)を削除すると, ペア(t, c_1)が新たにブロッキングペアになる可能性がある. これを防ぐため, 図 4 に示すアルゴリズム DELETE_SAFELY によってペアを削除する. DELETE_SAFELY では, ペア (t, c_1)をブロッキングペアとしてしまうようなペア(t_1, c_1)をさらに Q から削除することで, (t, c_1)がブロッキングペアとなることを防いでいる. なお, (t, c_1)が Q から削除されることで, c_1 は, t より選好することが既知である請負人とペアになることができる. そのため, 追加で削除されたペアは (t, c_1)はブロッキングペアとならない.

なお, ペア(t, c)が B_3 や B_4 に含まれる時には, 同順位を考慮してペアを削除する条件が変わるが, ほぼ同様である. また, これらは, 請負人の選好に注目して同様の処理を行う必要がある.

```
function MATCHING()

for each (B1, B2, B3, B4)
      s.t. |B1|+|B2|/2 + |B3|/2 + |B4|/4 = K/4
   Q = all pairs
   for each (t, c) ∈ B
      delete (t, c) from Q
      if (t, c) ∈ B1 ∪ B2
         for each c1 s.t. t prefer c1 to c
               or it is unknown if t prefer c1 or c
            call DELETE_SAFELY(c1, t, c, B, Q)
         end for
      else // if (t, c) ∈ B3 ∪ B4
         for each c1 s.t. it is known if t prefer c1 or c
            call DELETE_SAFELY(c1, t, c, B, Q)
         end for
      end if
      if (t, c) ∈ B1 ∪ B3
         for each t1 s.t. c prefer t1 to t
               or it is unknown if c prefer t1 or t
            call DELETE_SAFELY(t1, c, t, B, Q)
         end for
      else // if (t, c) ∈ B2 ∪ B4
         for each t1 s.t. it is known if c prefer t1 or t
            call DELETE_SAFELY(t1, c, t, B, Q)
         end for
      end if
   end for
   if there is stable matching M and |M| = n
         return matching
   end if
end for
if there is not stable matching
   return "there is no matching which PM(B) = K/4"
```

図 3： アルゴリズム MATCHING

このようにいくつかのペアが取り除かれたペアの集合 Q を用いて，安定マッチングを求める．ペアの集合 Q には未知の選好が含まれるが，これは同順位を許す安定結婚問題と同様の形式で表わしているため，形式的には，同順位を許す安定結婚問題と同様の安定マッチングを考えることができる．もしマッチングに未知の選好の選好を含むブロッキングペアが存在すると，初めに定めた B 以外の，新たなブロッキングペアとなる可能性があるが，これは，同順位を許す安定結婚問題の，同順位を含むブロッキングペアに相当する．そのため，ここでは超安定マッチングに相当するマッチングを求める必要がある．

また，Q からはいくつかのペアが取り除かれているが，これはペアになることを拒否することと同等である．これにより，必ずしも全員が含まれる安定マッチングが存在するとは限らなくなる．しかし，もしマッチされていないタスクと請負人がいる場合，初めに定めた B 以外の新たなブロッキングペアとなってしまう．そのため，全員がマッチングに含まれることが必要である．

これらより，Q から安定マッチングを求めることは，同順位とパートナーになることの拒否の両方を許す安定結婚問題において，全員が含まれる超安定マッチングを求めることに相当する．これは，Irving の超安定マッチングを求めるアルゴリズム[Irving 1994]に，パートナーになることの拒否を単純に加えることで求めることができる．そして，全員が含まれる超安定マッチングが存在すれば，そのマッチングは，ちょうど B のみをブロッキングペアとするマッチングとなる．

このペアの集合 Q からペアを削除し，安定マッチングを求め

```
function DELETE_SAFELY(x1, y, x, B, Q)

delete (x1, y) from Q
if (x1, y) is not in B
      and y prefer x1 to x
   for each y1 s.t. x1 prefer y to y1
      or it is unknown if x1 prefer y or y1
      delete (x1, y1) from Q
   end for
end if
```

図 4: アルゴリズム DELETE_SAFELY

るプロセスを，$E_M = K/4$ となる B それぞれについて行う．どれかの B に対して安定マッチングが存在すれば，そのマッチングは $E_M = K/4$ を満たす．一方，すべての B に対してそのようなマッチングが存在しなければ，$E_M = K/4$ を満たすマッチングは存在しない．このようにして，$E_M = K/4$ を満たすマッチングが存在するかを判定することができ，存在するならばそのマッチングを得ることができる．

このアルゴリズムを，$K = 0, 1, 2, \ldots$と順番に実行すると，最初に得られた安定マッチングを最大安定度マッチングとして得ることができ，ブロッキングペアの期待値 E_M は $K/4$ となる．

5. 結論

本研究では，大規模なタスク割り当て問題の解決の必要性から，未知の選好を含む安定結婚問題の定式化を行った．未知の選好を確率的に扱うことで，未知の選好を含む安定結婚問題を，ブロッキングペアの数の期待値の最小化の問題として表わすことができた．そして，期待値が最小化される最大安定度マッチングを求めるアルゴリズムを作成した．

このアルゴリズムでは最大安定度マッチングが求められることは保証されるが，計算量などの評価はまだ行っていないため，今後の課題となっている．このアルゴリズムの計算量のボトルネックとなるところは B を列挙するところであり，実際の問題に適用すると計算時間が現実的でないことは容易に想像できる．しかし，すべての B を列挙せずに，ブロッキングペアになると考えられるペアのみを選択的に B とするヒューリスティックを導入できれば，計算量が減らせるかもしれない．アルゴリズムを評価し，アルゴリズムの改良や，ヒューリスティックの導入をすることを今後の課題としたい．

謝辞

本研究は，日本学術振興会科学研究費基盤研究(B) (22300052, 平成 22 年度～24 年度)の補助を受けた．

参考文献

[Gale 1962] Gale D and Shapley L: College admission and the stability of marriage, The American Mathematical Monthly, Springer-Verlag, 1962.

[Eriksson 2008] Kimmo Eriksson and Olle Häggström: Instability of matchings in decentralized markets with various preference structures, International Journal of Game Theory, Springer-Verlag, 2008.

[Irving 1994] Robert W. Irving: Stable marriage and indifference , Discrete Applied Mathematics , Elsevier Science, 1994.

[Abraham 2006] David J. Abraham, Péter Biró and David F. Manlove, "Almost Stable" Matchings in the Roommates Problem, LNCS 3879, Springer-Verlag, 2006