看護師勤務表作成問題に対するヒューリスティクスおよび厳密解法に基づくアプローチの現在

渡邉　真也，稲船　淳也

看護師勤務表作成問題におけるアプローチをヒューリスティクスに基づくアプローチと数理計画に基づくアプローチに大別し，解説を行う．ヒューリスティクスに基づくアプローチとしては，これまでの筆者らの成果を中心に重要なポイントとなる選択と新規個体の生成部に特に焦点を当て説明する．また，数理計画に基づくアプローチとしては近年，その性能の高さから注目を集めている分枝価格法を利用したアプローチについてできる限りわかりやすく解説する．

キーワード：看護師勤務表作成問題，ヒューリスティクス，分枝価格法

1. はじめに

看護師勤務表作成問題 (Nurse Scheduling Problem, NSP)[1]は，スケジューリング問題の代表的な問題の一つであり，病院ごとに規則づけられた制約や希望のもとで，数週間から 1 カ月間といった一定期間の看護師勤務表を作成する問題である [1, 2].

NSP に対する社会的ニーズは高く，国内外において数多くのアプローチが提案され，大きな成果を挙げている [1, 3–7]. これらのアプローチは，遺伝的アルゴリズムなどのヒューリスティクスに基づくアプローチと分枝価格法や整数計画法といった数理計画に基づくアプローチに大別することができ，本稿ではこれら二つのアプローチに関する最新のアプローチについて，筆者らの研究成果も交えながら解説を行う．ヒューリスティクスに基づくアプローチに関しては，選択と新規個体の生成部のメカニズムに焦点を当てた解説を行い，数理計画に基づくアプローチに関しては，近年，注目を集めている分枝価格法を利用したアプローチについて解説を行う．

本稿の構成について述べる．まず，2 節において NSP についての簡単な概説を行い，3 節で Nottingham 大学において公開されている NSP に関するベンチマーク [8] の紹介を行う．続く 4 節においてヒューリスティクスアプローチに基づく方法について説明し，5 節で数理計画に基づくアプローチについて述べ，最後にまとめを述べる．

2. 看護師勤務表作成問題

ここでは，NSP の定義や特徴について概説する．ただし，ここでは一般的な内容の説明にとどめ，論文ごとの問題定義の異なり，設定の詳細については，サーベイ論文である文献 [1] に譲る．

一般に NSP は，m 人の看護師の 1 カ月（ここでは日数を n とする)[2]の勤務表をできる限り制約を満たすように作成することが目的であり [2, 5, 9, 10]，勤務形態を要素としてもつ $m \times n$ 行列を決定する問題として捉えることができる [5]. この勤務形態は，2 交代制では夜勤と日勤，3 交代制では準夜勤，深夜勤，日勤（早出，遅出）などのように分類されることが多い．

NSP における制約には，その制約を充足しないと勤務表として成り立たないハード制約とできる限り満たしてほしい（必ずしも満たす必要のない）ソフト制約の大きく 2 種類があり，NSP ではすべてのハード制約を満たした勤務表を生成することが第一の目標となる．以下，代表的な制約を示す．

- 勤務表の縦方向に関する制約条件（看護の質に関する制約）
  - ・各日の各勤務形態に必要な人数
  - ・グループ人数

わたなべ　しんや，いなふね　じゅんや
室蘭工業大学大学院しくみ情報系領域
室蘭工業大学大学院情報電子工学系専攻
〒 050–8585 室蘭市水元町 27–1
sin@csse.muroran-it.ac.jp
12024017@mmm.muroran-it.ac.jp

[1] Nurse Rostering Problem (NRP) とも呼ばれる．
[2] 国内では 1 カ月が基本の期間単位となっているが海外では 2 週間，3 週間といったように週間で区切る設定が多い [8].

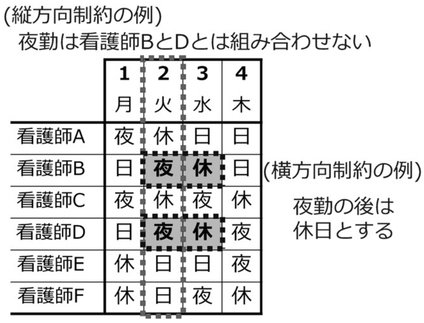

図 1　NSP における制約同士が干渉する一例

- 勤務表の横方向に関する制約条件（看護師の生活の質に関する制約）
  - ・各看護師に割当てる勤務回数
  - ・勤務パターン

NSP における制約条件は看護の質に関するものと看護師の生活の質に大別することができ，前者の多くが勤務表における縦方向（日付ごとの制約）に関する制約であり，後者の多くが横方向（スタッフごとの制約）に関する制約であることからそれぞれ縦方向制約と横方向制約として捉えることができる．

NSP における最大の特徴は制約条件の複雑さとその充足の難しさである．制約同士の干渉に関する具体的な例を図 1 に示す．この図では，縦方向制約である看護師 B および C の夜勤での組み合わせ禁止と横方向制約である夜勤の後は休日とする勤務パターン制約がバッティングしている例を示している．NSP では，この例に示すようにある制約条件を充足させる操作によりほかの制約条件が違反してしまうような状況を招きやすく，特に，探索が進んだ解（勤務表）ほどこの相互干渉が生じやすくなるという特徴を有している．

多くの NSP では，一定期間の勤務表そのものを最適化パラメータとして扱い，制約違反数を最小化する問題として NSP を定式化する．

ここでは，看護師の人数を m，日数を n とした $m\times n$ 行列（式 (1)）を解として表現し，行列の各要素には式 (5) に示す s 種類の勤務形態の中の一つが必ず割り当てられているものとする．また，制約違反を縦方向の勤務制約条件と横方向の看護師制約条件に分けて考えた場合における目的関数の設定例を式 (6) に示す．式 (6) の $f^{\mathrm{hospital}}(X)$, $f^{\mathrm{people}}(X)$ は式 (7)，式 (8) に対応しており，それぞれ「看護の質に関わる制約」,「看護師の生活の質に関わる制約」の違反数から算出される評価値を表している．

$$\underset{m\times n}{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{p1} & x_{p2} & \cdots & x_{pn} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix} \tag{1}$$

$$x_{pq} = w_r \quad (p \in M, q \in N, w_r \in W) \tag{2}$$

$$M = \{1, 2, \ldots, m\} \tag{3}$$

$$N = \{1, 2, \ldots, n\} \tag{4}$$

$$W = \{w_1, w_2, \ldots, w_s\} \tag{5}$$

$$\mathrm{Minimize} f(X) = f^{\mathrm{hospital}}(X) + f^{\mathrm{people}}(X) \tag{6}$$

$$f^{\mathrm{hospital}}(X) = \sum_{i=1}^{k} \omega_i^{hospital} c_i{}^{hospital}(X) \tag{7}$$

$$f^{\mathrm{people}}(X) = \sum_{i=1}^{l} \omega_i^{\mathrm{people}} c_i{}^{\mathrm{people}}(X) \tag{8}$$

それぞれの制約には，優先度の高さに基づいた重み (ω) が設定されており，優先度の高いものほど大きな重みが設定されている．式中における記号定義を以下に示す．

k: 勤務制約条件（縦方向制約条件）の数
l: 看護師制約条件（横方向制約条件）の数
c_i: 制約条件 i の違反数
ω_i: 制約条件 i の重み．

3. ベンチマーク問題について

NSP に対するアプローチ比較の大きな問題の一つに，扱う問題が共通化されていないため，正確な比較が行えないという問題がある．これは，多くの NSP 研究がそれぞれ直面する現場に対応したアルゴリズムの設計を行っており，現場で求められるシビアさを例題に求めるほど固有の条件設定が必要となり，結果として扱う対象問題の制約条件設定および種類が異なってしまうことに起因する．

この問題に対して Nottingham 大学の Curtois らのグループは主要論文において扱われた例題を収集するとともに，独自の例題を作成し，Employee Shift Scheduling Benchmark Data Sets として自身の Web サイト上から広く公開している [8]．このサイトでは，2016 年現在，彼らオリジナルの例題が 24，ほかの主要論文で利用されてきた 25 例題（うち 21 例題では，既知の最良解 (Best known solution) に関する詳細な情報が示されている）が公開されており，近年では彼らのグループを中心にではあるものの，手法の有用性を検証するためにこれらのベンチマークを利用した事例も発表さ

れている [6].

このサイトでは単に問題の詳細が示されるだけではなく，最良解の情報も常に更新されるなど手厚いサポートが実施されており，今後の NSP アプローチ開発ではこれらのベンチマーク問題に対する数値実験が必須となっていくものと思われる．

4. ヒューリスティクスに基づくアプローチ

後述する厳密解法のアプローチに比べ，より大規模な問題においても適用可能なヒューリスティクスは，NSP において古くから数多くの事例が紹介されており，遺伝的アルゴリズム (Genetic Algorithm, GA) や差分進化法 (Differential Evolution, DE)，タブーサーチおよびそれらを組み合わせたものなど多種多様なアプローチが提案されている [5, 10–12].

ヒューリスティクスに基づくアプローチの特徴の一つとして，NSP を多目的問題として扱う試みが挙げられる．この NSP の多目的化は，主に多様な解候補の導出を目的として行われているが，ヒューリスティクスと多目的最適化との親和性の高さも大きな要因となっている．たとえば，Berrada et al. は NSP を 4 目的最適化問題として捉え[3]，Sequential Method，均等の重みを利用した方法，タブーサーチに基づく方法など異なる三つのアプローチによる実装およびそれらの比較実験について報告している [11]．その他，Cai and Li は GA においてパレートランクに基づく選択を採用した手法を提案しており，交叉後の個体に対して制約違反を改善するためのメカニズムを組み込むなど効率よく多様な解候補を導出することを目的としたアルゴリズムとなっている [12].

このように NSP に対して単目的，多目的それぞれの枠組みで独自に工夫したヒューリスティクスを用いた数多くの試みがなされているが，ここでは筆者らの NSP に対する取り組み [7, 13, 14] から得られたヒューリスティクスアプローチにおいて効果的な探索を実現するためのキーとなる部分に焦点を絞って解説する．

4.1 効果的なヒューリスティクスアプローチ実現のために

一般的なヒューリスティクスアプローチの流れを図 2 に示す．終了判定の場所や新規個体生成と選択が多段階的になっているなどの違いはあるものの，すべてのアプローチが原則的には，図 2 の流れに従う．

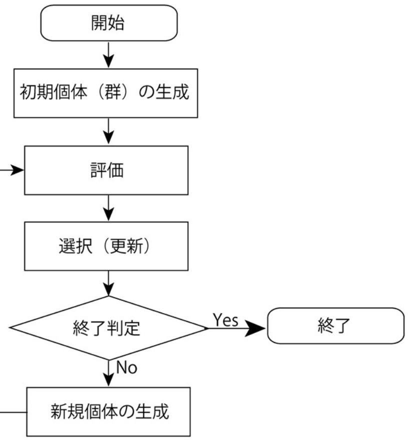

図 2 ヒューリスティクスアルゴリズムの一般的な流れ

この流れにおいて探索に特に重要な役割を果たすのが選択と新規個体の生成部である．選択には，それまでの探索において発見した良質な解情報を保存する役割と，次期の新規個体を生成する際の有望な種となる解を選択する役割の大きく二つの役割がある．筆者の個人的な見解では，探索効率の優れないアプローチの多くが，選択においてこれら二つの役割をメカニズムの中に適切に組み込めていないように思われる．

一方，局所解からの脱出が大きな鍵を握る NSP において，いかに探索の停滞を防ぎ，現在，未充足の制約を充足できる可能性がある（そのような解へ辿りつくような）次世代の個体を生成できるかは探索効率を左右する重要なポイントとなる．

ここではこれまでの筆者らの取り組み [7, 13, 14] に基づき，複数の個体群を扱うヒューリスティクスアプローチを前提とした選択と新規個体の生成に関するメカニズムのポイントを述べる．

4.2 選択メカニズムの設計ポイント

ヒューリスティクスアプローチにおける選択には，良質な解を保存することと次の新たな個体を生成するための種を選ぶことの二つの役割がある．便宜上ここでは，前者の選択を記録するための選択という意味でアーカイブ選択と呼び，後者を探索を行うための選択という意味で探索個体選択と呼ぶ[4].

両者の選択は互いに深く依存し合っており，アーカイブ選択で選出した個体群の中から次の個体を生成するための種となる個体群を探索個体選択で選択するという関係にある．そのため，アーカイブ選択の段階にお

[3] 彼らの論文では，まず NSP を 7 目的最適化問題として捉え，目的数の削減のためそれらを四つのカテゴリーに分類し直し，4 目的問題として実験を行っている．

[4] 進化型多目的の分野では，前者をアーカイブ選択，後者をメイティング選択と呼び 2 段階の選択を実現している [15].

いて単に評価値の高さだけで選択するのではなく，有望なポテンシャル（将来的にさらなる制約充足の実現につながる可能性）をもつ解を選ぶことがポイントとなる．

筆者らの文献 [7] におけるアプローチでは，新規生成した個体がその生成元となった個体に対して，例え一部の制約違反に関してでも優位性をもつ場合には積極的にアーカイブ選択時に選択する方針をとっており，制約充足度合いを表す評価値と同様，いかに部分的に優れているかについても陽に考慮する戦略をとっている．

また，文献 [13] におけるアプローチでは，探索の充足が難しい制約の重みを動的に変化させることによる大幅な性能向上について報告している．本アプローチでは，最良解における各制約充足具合を一定世代間隔でモニタリングし，改善が見られない制約の重みをその時点の 2 倍に設定するという操作を行っている．各選択操作は，変化させた重みで算出し直した評価に基づいており，充足が困難な制約違反が優先的に解消されやすくなる設定となっている．

4.3 新規個体生成メカニズムの設計ポイント

NSP は局所解を多数有する多峰性の問題であるため，新規個体を生成する際には探索の状況に応じて下記の二つの戦略を適切に使い分ける必要がある．

局所探索：現在の解の周辺を調査し，より良質な解の生成を試みる．

局所解脱出：一時的な解の改悪を許容し，現在の解の周辺から離れることを試みる．

具体的には，通常は最良解もしくは良質と思われる解を種にその周辺を探索する局所探索を行い，探索が一定期間停滞（最良解の更新がない）しており明らかに局所解に陥っていると判断したときに局所解脱出を行い，現在の解が存在する領域から離れるというメカニズムを組み込む必要がある．

上記の二つの戦略について万能な方法は存在しないものの，これまでの筆者らの取り組み [7, 13, 14] では，無駄な探索の重複を防ぐタブーサーチには明らかな局所探索の性能向上が認められたため，局所探索に何らかの探索履歴情報を活用するメカニズムを組み込むことを薦める．

また，局所解脱出において重要なのは，いったん，現在保存している個体の情報をすべて破棄し，新規に生成した個体情報の情報のみで探索を再開することである[5]．これは，新規に生成した個体の評価値は元の個体に比べ大きく悪化している可能性が高く，元の個体群の情報が残っている場合には上述の選択操作においてそれらの個体が選択され，元の局所解に再び戻る危険性があるためである．

5. 数理計画に基づくアプローチ

これまで分枝限定法といった厳密解法に分類されるアプローチを NSP へ応用した事例は数多く存在する [3, 6, 16, 17]．その中でも，Maenhout and Vanhoucke のグループによりその効果が広く知れわたった分枝価格法に基づくアプローチが脚光を浴びており [16], Burke and Curtois は分枝価格法適用の際にネックとなる被約費用最小化問題 (Princing problem, PP) を高速に処理するアプローチにより驚異的な成果を報告している [6].

この分枝価格法は NSP において強力かつ，効果的なアプローチであることが明らかとなっているものの，必ずしもその理解は容易ではなく，実装には数学的な知識とある程度のプログラミングスキルが要求される．ここでは，文献 [6, 16] の取り組みを参考に NSP に対する分枝価格法の適用について，少なくとも概念的に理解できるよう心がけた解説を行う．そのため，詳細な数理モデル定式化の部分などについては，これらの文献 [6, 16] を参照されたい．

5.1 分枝価格法

分枝価格法は，分枝限定法（列挙法）と列生成 [18, 19] を組み合わせて厳密解を求めるための方法論であり，1990 年代後半から注目を集め，割り当て問題全般に広く応用されている [20].

直感的には，分枝限定法の各部分問題に対して列生成法を適用し，そこで得られた情報をもとに限定操作，分枝操作を行う方法として分枝価格法を理解することができるが，実際には，列生成法を通して最適な整数解を得るために列挙法の一つである分枝限定法を利用している．すなわち，列生成法により得られた実数解の一部を整数値に置き直す形で分枝操作を行い，最終的な整数解の導出を実現している．

以下，分枝価格法を理解するための肝となる列生成法について簡単に述べた後，NSP に対する具体的な適用方法，このアプローチにおける課題の順に述べる．なお，分枝価格法の詳細については，文献 [6, 16, 20] が参考になる．

5.2 列生成法

列生成法は，有望な解の部分集合を部品として列挙し，その被覆問題として定式化することで解を求める

[5] 記録のため最良解の情報保存する際には，探索には利用しない形で保存する必要がある．

アプローチである [21]．可能なすべての解候補をあらかじめ列挙するのではなく，有望な解候補を必要に応じて作成することにより，NSP のように変数や制約が多い問題に対しても効率よく厳密解を得ることができる．

直感的な理解のために，主要な部分だけを抜き出した列生成法の枠組みを下記に示す．

Step A：いくつかの解候補を用意（初期の列集合）し，それをもとに候補集合が限定された被覆問題である限定主問題 (Restricted Master Problem, RMP) を作成する．

Step B：RMP に対する線形緩和問題を作成し最適解および双対変数を得る．

Step C：線形緩和問題の双対問題を作成し，Step B で得られた最適解の情報を利用して最適解が満たしていない双対問題の制約式を探索する[6]．

Step C-1：そのような制約式がなければ（すべての制約式における余裕がない）最適解が得られたものとして終了する．

Step C-2：そうでない場合，その制約式に対応する主問題の変数を Step A における RMP に追加し，Step B へ移る（列の追加操作）．

上記の列生成法を適用することにより，線形緩和問題の最適解と最適値を得ることができる．この最適値こそが元の問題における下界値となり，分枝限定法における限定操作に利用することができる．また，最適解の情報からどの変数を固定して分枝すればよいかという情報を得ることができる．

列生成法における特徴は，問題を集合被覆問題における集合セット（解候補）が限定された RMP として定式化し，実際に最適解を求めるのはその線形緩和問題である点，そこで得られた緩和問題に対する最適解の情報から現在の解の厳密性の検証および追加すべき新たな列生成（解候補）のための情報を得ている点である．これらの特徴により，多くの変数，制約をもつ大規模な問題において効率的な厳密解の導出が可能となっている．

ここで，NSP に対して列生成を適用する場合を考える．この場合，列生成における列は各看護師のその期間のスケジュールとなり，各看護師におけるスケジュール候補を次々に作成することで候補集合を充実させ，最終的にはその候補集合の情報からその看護師における最適なスケジュールを求めることになる．

[6] 実際には，線形緩和問題における被約費用最小化問題として定式化し，その最適値の正負で判断する．

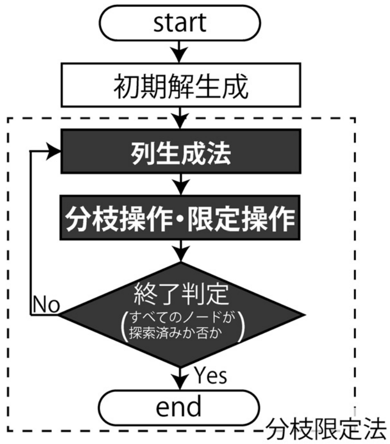

図 3　NSP に対する分枝価格法適用の流れ

5.3　NSP に対する分枝価格法適用の手順

NSP に対して分枝価格法を適用する手順を示す前に，関連する重要語句の説明を下記に示す．

双対価格 (Dual Cost)　シンプレックス法で RMP を解いた際に導出される双対問題の最適解．

被約費用 (Reduced Cost)　Dual Cost と 1 人の看護師のスケジュール（列）の目的関数値（横制約のみ）から算出される値（負の値であれば，RMP の解を改善する見込みのある列だと判断される）であり，上述の双対問題における制約の余裕度合いを表す．

列プール F_i　看護師 i のためのスケジュール集合．各看護師のスケジュールが列に該当する．

NSP に対する分枝価格法適用の流れ図を図 3 に示し，図中で示した各手順の詳細を以下に示す．

Step 0) 初期解生成． 解 X をランダムに生成し，上界値 (Upper Bound, UB) $= f(X)$，下界値 (Lower Bound, LB) $= f(X)$ とし，列プール F_i に各看護師 (i) のスケジュールを追加して，Step 1) へ進む．

Step 1) 列生成法． 分枝限定法における部分問題の線形緩和問題を解く．

Step 1-1) RMP を解く． シンプレックス法を適用し，列プール F 内の列を変数とした線形緩和解 X' を生成し，LB $= f(X')$ とする．この際，新規列を生成するために使用される双対価格が算出され，X' が整数解の場合，上界値も同時に更新する．

Step 1-2) 新規列を生成． 被約費用最小化問題 (Princing Problem, PP) の解として看護師 (i) の新規スケジュール（新規列）を生

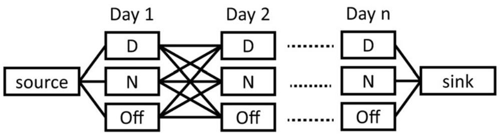

図 4　多段ネットワークにおける最小経路問題の例

成する.

Step 1-3) 列生成の終了判定. 列プール F_i に負の被約費用を有した列 C_i' を追加する. 列プールに更新があれば, Step 1-1) へ戻り, そうでない場合, RMP が最適に解けたと判断し, Step 2) へ進む.

Step 2) 分枝限定操作. $UB > LB$ の場合, 分枝操作を適用, $UB \leq LB$ の場合, 限定操作を適用する.

Step 3) 探索の終了判定. Step 2) の操作により生成された木構造のすべてのノードが探索済みと成った段階で探索終了. そうでない場合, Step 1) へ戻り再度部分問題を解く.

このように分枝価格法は, 分枝限定法における分枝・限定操作に列生成法を利用する枠組みに基づいており, 列生成法における線形緩和解導出にシンプレックス法, 新たな新規列の生成のための PP 最適化には動的計画法が用いられている.

NSP に分枝価格法を適用する際に大きな問題となるのは探索効率である. 筆者らが文献 [16] を参考に分枝価格法を NSP に適用した結果, 看護師数が限定されている, もしくはスケジュール日数が比較的短い問題ではすぐに厳密解が導出できたものの, 看護師数が 30 名を超える規模の問題では短時間で解を得ることはできなった.

分枝価格法を NSP に適用する際, 筆者が考える探索効率を左右するポイントは下記の 3 点である.

・「Step 1-2) 新規列の生成」において, いかに効果的な列を正確かつ高速に算出できるか.
・「Step 2) 分枝限定法の適用」において, 生成した木構造をどのような順番で走査するか.
・「Step 2) 分枝限定法の適用」 の分枝操作をどのように行うのか.

一つ目の問題である列生成部分について, Burke and Curtois は非常に効果的なアプローチを提案し, 驚異的な成果を報告している [6]. 彼らは, 看護師一人のスケジュールを図 4 に示すような多段ネットワーク（各層は一日分の勤務を表し, この例では各日において Day/Night/Off の 3 ノードのいずれかが選択されるようになっている）として表現し, その最短経路を求める形で PP における最適勤務を決定している. この多段ネットワークにおける最適解は動的計画法（ダイクストラ法など）により算出することができるが, 多段ネットワークのアークの重みが限定主問題の双対最適値に基づき設定されていること, 実行不可能なアークは事前に削除 (prune) されていることなどから, 非常に高速に解（新規の列）を導出することに成功している.

二つ目の問題である木構造探索には, 深さ優先探索, 幅優先探索といった一般的な方法から, 有望と思われるノードを優先して探索する最良優先探索, 深さを段階的に増やして探索する反復深化深さ優先探索まで多種多様な探索アルゴリズムが存在する. しかしながら, どのアプローチが適切であるかは対象となる木構造の形, 深さ, さらには優先度の適切な評価を設定できるかどうかという点に依存するため最適な探索を見つけることは容易ではない. ただし, どのノードから順に探索を行うかは探索効率に大きく影響するため, 十分に吟味して実装する必要がある.

また, 三つ目の問題である分枝操作も探索効率に多大な影響を与える重要な問題である. 分枝価格法を NSP に適用する場合, 分枝を行う基となる情報はシンプレックス法の適用により得られた線形緩和解となる. 得られた線形緩和解の情報から分枝を行うシンプルな方法としては, 0.5 といった整数 (0 or 1) の中間に近い値をとる変数を基準に分枝を行う方法, もしくは整数値に近い値をとる変数（たとえば最も 1 に近い変数）を基準に行う方法の二つが考えられるが, Burke and Curtois によると後者のアプローチのほうが性能が高いと報告されており [6], 実際に筆者らがこの 2 種類の比較実験を行ったところ彼らの記述どおり後者のほうが探索効率がよいことが確認できた. また, Maenhout and Vanhoucke は分岐の形状と分枝基準の関係について詳細に比較しており, 2 分岐, 多分岐, 制約つき分岐（3 分岐）といった異なる三つの分岐方法と上述の 2 種類の分枝基準を組み合わせた場合の比較実験について報告しており, 整数の中間値に近い値をとる変数を基準に 2 分岐する方法が最も優れているとの結論を示している [16].

分枝価格法は NSP に対して効果的なアプローチであることは間違いないが, 上述の問題点を十分に考慮しなければ期待した探索効率を得ることができない. 十分に探索効率を向上させた実装でない場合, 単に探索に時間がかかるというだけでなく, 適用可能となる

問題が小規模のものだけに限定されてしまうため注意する必要がある．

6. おわりに

本稿では，NSP に対するアプローチをヒューリスティクスに基づくアプローチと数理計画に基づくアプローチに大別し，解説を行った．

今後の NSP アルゴリズムでは，分枝価格法の存在がますます大きくなり，より大規模な問題への分枝価格法の適用といった課題が重要なトピックになると思われる．そのため，それらの課題解決を目的とした，ヒューリスティクスとのハイブリッド化や効率的な新規列の生成法の開発といった研究の進展が期待される．

また，高速な解導出が可能となった今，ユーザーと対話形式でスケジュール作成支援を行うシステム，代替案について積極的に提案するシステムなど新たな機能を持つアプローチも登場してくるものと思われる．

参考文献

[1] E. K. Burke, P. De Causmaecker, G. V. Berghe and H. Van Landeghem, “The state of the art of nurse rostering,” *Journal of Scheduling*, **7**, pp. 441–499, 2004.

[2] 池上敦子，“ナース・スケジューリング—調査・モデル化・アルゴリズム—，” 統計数理，**53**, pp. 231–259, 2005.

[3] G. Baskaran, A. Bargiela and R. Qu, “Integer programming: Using branch and bound to solve the nurse scheduling problem,” In *Proceedings of 2014 International Conference on Artificial Intelligence and Manufacturing Engineering*, pp. 203–209, 2014.

[4] B. Jaumard, F. Semet and T. Vovor, “A generalized linear programming model for nurse scheduling,” *European Journal of Operational Research*, **107**, pp. 1–18, 1998.

[5] 串田淳一，大場和久，亀井且有，“Differencial Evolution によるナーススケジューリングアルゴリズムの提案，” 進化計算シンポジウム 2011, pp. 299–306, 2011.

[6] E. K. Burke and T. Curtois, “New approaches to nurse rostering benchmark instances,” *European Journal of Operational Research*, **237**, pp. 71–81, 2014.

[7] 渡邉真也，奥寺将至，“看護師勤務表作成問題に対する進化型多目的最適化に基づくアプローチの提案，” 進化計算学会論文誌，**5**, pp. 32–44, 2014.

[8] T. Curtois, Employee Shift Scheduling Benchmark Data Sets, http://www.cs.nott.ac.uk/~psztc/NRP/

[9] 新妻真輔，池上敦子，品野勇治，“列生成法を用いたナーススケジューリング問題の解法，” 数理モデル化と問題解決研究報告　情報処理学会研究報告 (MPS)，**2009**, pp. 221–224, 2009.

[10] 川中普晴，山本康高，吉川大弘，篠木剛，鶴岡信治，“遺伝的アルゴリズムを用いた看護婦勤務表の自動生成，” 電気学会論文誌，**122-C**, pp. 1023–1032, 2002.

[11] I. Berrada, J. Ferland and P. Michelon, “A multi-objective approach to nurse scheduling with both hard and soft constraints,” *Socio-Economic Planning Sciences*, **30**, pp. 183–193, 1996.

[12] X. Cai and K. N. Li, “A genetic algorithm for scheduling staff of mixed skills under multi-criteria,” *European Journal of Operational Research*, **125**, pp. 359–369, 2000.

[13] J. Inafune and S. Watanabe, “The effectiveness of considering the intrinsic characteristics for nurse scheduling,” In *Proceedings of Joint 8th International Conference on Soft Computing and Intelligent Systems and 17th International Symposium on Advanced Intelligent Systems (SCIS& ISIS 2016)*, 2016.

[14] 奥寺将至，渡邉真也，“看護師勤務表作成問題における満足解導出後の解洗練アプローチの提案，” 第 10 回進化計算学会研究会資料集，2016.

[15] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, Wiley, 2001.

[16] B. Maenhout and M. Vanhoucke, “Branching strategies in a branch-and-price approach for a multiple objective nurse scheduling problem,” *Journal of Scheduling*, **13**, pp. 77–93, 2010.

[17] J. F. Bard and H. W. Purnomo, “Preference scheduling for nurses using column generation,” *European Journal of Operational Research*, **164**, pp. 510–534, 2005.

[18] M. V. D. Akker, H. Hoogeveen and S. V. D. Velde, “Combining column generation and Lagrangean relaxation: An application to a single-machine common due date scheduling problem,” *Informs Journal on Computing*, **14**, pp. 37–51, 2000.

[19] J. Desrosiers and M. E. Lübbecke, “A primer in column generation,” *Column Generation*, G. Desaulniers, J. Desrosiers and M. M. Solomon (eds.), Springer, pp. 1–32, 2005.

[20] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh and P. H. Vance, “Branch-and-Price: Column generation for solving huge integer programs,” *Operations Research*, **46**, pp. 316–329, 1996.

[21] 宮本裕一郎，“はじめての列生成法，” オペレーションズ・リサーチ：経営の科学，**57**(4), pp. 198–204, 2012.

