

ナース・スケジューリング

— 調査・モデル化・アルゴリズム —*

池上 敦子[†]

(受付 2005 年 5 月 13 日; 改訂 2005 年 6 月 9 日)

要 旨

日本における病院の数は 1 万を超すとも言われ、毎月約 3 万部署でナース勤務表作成の仕事が発生していると考えられる。このスケジューリング問題は、看護の質を守るとともにナースの労働負担を十分考慮しなければならないことから、非常に多くの種類の制約が存在し、解くことが困難な組合せ問題となっている。この論文では、現場調査に基づくモデル化から適用までの「問題解決の全過程」を紹介することにより、本研究の最近の成果を報告する。先ず、ナース・スケジューリングを数理計画モデルで記述することにより、問題の拘束条件マトリックスがブロック対角構造を持つことを明らかにする。そして、その構造を利用した「部分問題軸アプローチ(Subproblem-centric Approach)」という考え方を提案し、これに基づくアルゴリズムで問題を解く。アルゴリズムは、実際の病棟の問題に対し非常に質の良い勤務表を効率良く与えることができた。提案するモデルとアプローチは、海外のナース・スケジューリングはもちろん、医療、介護、サービス業等の中でも、勤務の質が特に要求されるようなスタッフ・スケジューリングに適用可能である。

キーワード： ナース・スケジューリング，アンケート調査，数理計画モデル，ブロック対角構造問題，タブサーチ。

1. はじめに

医療の現場では、人の命に関わる業務をおこなっていることから、そのスタッフらで構成される勤務の質が非常に厳しく求められている。特に、24 時間切目ない活動が要求される病棟ナースの勤務表作成(ナース・スケジューリング)では、看護の質とスタッフ・ナースの生活の質の両方を守るべきことから、全ての条件を満たす勤務表の作成が難しい。ナース・スケジューリングの難しさは国内外で広く知られているが、日本においてはスケジューリング・アルゴリズムの利用がまだ一般的ではなく、紙面での作成や、表計算が可能な程度のソフトウェアの利用が主流である。また、勤務表作成者のプライベート時間も含めた非常に多くの時間と労力が費やされているのが現状である。1997 年の国内 23 病院の 315 部署に対する我々の調査(池上, 2001)では、考えている時間を含めると、1 部署 1 ヶ月分の勤務表を作成するのに平均約 11 時間も費やされていることがわかっている。その後も多くのソフトウェアが出回っているものの、現場から勤務表作成の困難を訴える声が減る様子は見られない。

[†] 成蹊大学 理工学部：〒180-8633 東京都武蔵野市吉祥寺北町 3-3-1

* 本稿は、Ikegami and Niwa(2003)ならびに、池上(2004)の内容に対して一部修正・加筆したものである。

勤務表は通常、その部署の看護師長が主任が作成するが、スタッフ・ナースの労働負荷や、休み希望といった個々の要望を考慮しながら、必要な人数とスキルレベルのナースを各シフト(2 交替制では日勤と夜勤, 3 交替制では日勤と準夜勤と深夜勤)に割当てなければならない。しかし、退職者(特にベテラン)が多く出てしまう下では、これを達成することが困難となり、その結果好ましくないシフトの並びが作成されてしまうと、ナースの睡眠パターンや健康にも悪い影響を及ぼし、また退職者が増えてしまうという悪循環が起こる。このような問題を少しでも解決できるよう、効率のよいスケジューリングのアルゴリズムの研究開発が必要とされてきた。

このような研究は 1970 年代からすでに始まっていたが(サーベイ論文 Li and Benton, 1996; Siferd and Benton, 1992; Smith-Daniels et al., 1988 参照), 多くのものはスケジューリング期間を 1 週間または 2 週間のブロックに分けて扱っていたり(Arther and Ravindran, 1981; Bell et al., 1986; Chan et al., 1987; Jaumard et al., 1998; Kostreva and Jennings, 1991; Megeath, 1978; Millar and Kiragu, 1998; Miller et al., 1976; Rosenbloom and Goertzen, 1987; Warner, 1976), 初めからスケジューリング期間を 1 週間か 2 週間と定義して扱っている(Aickelin and Dowsland, 2000; Dowsland, 1998; Dowsland and Thompson, 2000; Jaumard et al., 1998; Musa and Saxena, 1984; Ozkarahan and Bailey, 1988)。また、これまでに 3 交替を扱ったどの論文(Arther and Ravindran, 1981; Chan et al., 1987; Jaumard et al., 1998; Kostreva and Jennings, 1991; Megeath, 1978; Meyer auf'm Hofe, 2001; Ozkarahan and Bailey, 1988; Randhawa and Sitompul, 1993; Warner, 1976)も、2 交替を扱ったどの論文(Aickelin and Dowsland, 2000; Bell et al., 1986; Dowsland, 1998; Dowsland and Thompson, 2000)も、限られた種類のシフトだけ働くナースを対象としていたり、1 種類のシフトが休日を含みながらも 1 週間も 2 週間も続くといった周期の長いローテーションを対象にしている。日本では、ほとんどのナースが週に 2~3 回シフトが変わるといった短い周期のローテーションで働いており、日本の病院に対し、既存のナース・スケジューリングのアルゴリズムを直接利用することは不可能であることがわかった。Millar and Kiragu (1998)だけは、短い周期の 2 交替の問題を扱っているが、通常 20 名以上 4 週間以上の大きさを持つ日本の問題に比べて、その問題サイズは小さい。

ナースは、一般にスキルレベルや担当する患者によってグループ分けされているが、ナース(特にベテラン)が足りない状況では、複数のグループにまたがる拘束条件が存在してしまい、グループ毎に独立なスケジューリングをすることができない。また、もっと短いスケジューリング期間(例えば 2 週間)では、個々のナースの希望を受け入れながらの負荷の平準化が難しい。結果として、その部署に所属する通常 20 名以上のナースそれぞれに 1ヶ月もしくは 4 週間分の、サイクリックでない別々のスケジュールを用意しなければならない。これまでの研究では、このような実際の制約を理論的に取り扱ってこなかったが、実際の制約を考慮できるナース・スケジューリングのモデルを構築することは非常に大事なことだと考える。また、日本以外のイギリスやイスラエルの病院にも、日本と同様なスケジューリング問題が存在していることにも注目する。本研究では、ナース・スケジューリング問題の本質に取り組む方法の開発を目指す。また、我々の最初の 2 つの論文、池上 他(1995, 1996)に続いて、日本でもナース・スケジューリングの研究がすでに始まっている(長野・宮崎, 1996; Nonobe and Ibaraki, 1998; Osogami and Imai, 2000, 他)。

この論文では、現場調査に基づくモデル化から適用までの「問題解決の全過程」を紹介することにより、我々の最近の成果を報告する。初めに、日本におけるナース・スケジューリング問題について述べてから、勤務表作成の現状把握のために我々がこなした現場調査の結果と、その結果に基づく数理計画モデル(池上 他, 1996)を紹介する。そして、このモデルで明らかになった構造をうまく利用した効率のよいアプローチ(池上・丹羽, 1998)を提案し、このアプローチにのっとり、効率よく適切な解を提供できる 2 交替用アルゴリズム(池上, 2000)を示す。さ

ており、それぞれスキルレベルでも 3 つのグループに分けられていた。表中の記号 ‘□’, ‘Nn’, ‘+’, ‘/’ は、それぞれ「日勤」、「夜勤」、セミナ等「その他の勤務」、「休み」を表すが、夜勤の ‘Nn’ は、2 日にわたる勤務である。表の下には、各シフトで働くナースの数、表の右には、各ナースがスケジューリング期間に働くシフトの数が与えられている。

勤務表作成者は、各シフトに働くナースのスキルレベルや相性、個々のナースの希望、さらに、健康に悪影響を及ぼすようなシフトの並びを避けたり、前スケジューリング期間からのシフトの並び等を考慮しながら、これらの数が適正な幅に収まるようにしなければならない。日本では、ローテーションの周期が短い(1 種類のシフトが長く続くことを好まない)ので、シフトの並びに対する制約が数多く存在する。

3. 現場調査

日本におけるナース・スケジューリングは、1994 年におこなった東京女子医科大学附属病院 40 部署でのアンケート調査及び聞き取り調査(池上 他, 1995)で初めて明らかになり、1997 年の国内 23 私立大学附属病院 615 部署中の 315 部署を対象におこなったアンケート調査(池上, 2001)で、その内容が確認されている。この章では、日本のナース・スケジューリング問題のモデル化のベースとなった、1 つ目の調査の結果を紹介する。

東京女子医科大学附属病院(1994 年当時ベッド数 1421)におけるアンケート調査の対象者は、病棟ナースの勤務表作成を担当している婦長 3 名と主任及び主任代理 37 名、計 40 名であった(婦長は現在は師長と呼ばれている)。勤務表作成歴は、平均 7 年(5 ヶ月~23 年)であり、担当部署の勤務体制は 3 交替制が 34 部署、2 交替制が 6 部署であった(1996 年に、国立病院・療養所が 2 交替制を導入したことにより、現在では 2 交替制を導入する病院も増え、東京女子医科大学附属病院でも 1998 年以降は半分以上の部署が 2 交替制になっている)。

勤務表のスケジューリング対象期間としては、夏休み、冬休みの時期を 2~3 ヶ月単位で作成することを除けば、アンケート回答者の 39 名が月初 1 日から月末までの 1 ヶ月単位でおこなっており、残りの 1 名はそれを 15 日の前後で分けた半月単位でおこなっていた。スケジューリング対象人数は、平均 24.3 人(12~42 人)であった。

アンケートは 29 項目からなり、勤務表作成に費やす時間、精神的負担、具体的作成手順、コンピュータ・システムに対する期待などを質問にした。ここでは、それらの結果を、勤務表作成の困難さと時間的負担、勤務表作成の手順、勤務表が満たすべき条件、望ましい勤務表、スケジューリング・システムについての意見、に分けて整理した結果を示す。

勤務表作成の困難さと時間的負担

勤務表を作成するために費す平均時間は 6.8 時間(40 名の平均)であり、ほぼ勤務 1 日分を費やしていることがわかった。そして、長くかかってしまうときには、30 時間も費す人達がいるなど、作成時間の数字を見ただけでも時間的負担はかなり大きいものと思われる。このことをもっと強調するのは次の結果である。勤務表作成においてプライベート(勤務終了後や休日)の時間が占める割合を調べた結果、勤務時間内だけで作成できるのが 2 名だけであったのに対し、全て勤務時間以外の時間を使っている人が 18 名もいた。この長い時間を要し、自分のプライベートの時間を割かねばならない勤務表作成に対しては、「楽しい」と感じている人が 2 名、「苦痛」に感じている人が 28 名であり、7 割の人が苦痛に感じていることになる。ここで、楽しいと感じる場合の勤務表作成平均時間は 3 時間と 4 時間であり、勤務時間以外の時間が占める割合も 0%、10%と非常に少ない。逆に、勤務表作成平均時間が 4 時間を超してしまうと、苦痛と感じる割合が、4 時間以内の場合の 2 倍以上(85%)になり、楽しいと感じる人もいなくなってしまう。

作成した勤務表に対するまわりの評価については、「苦心して作成しても文句が出る」と言いながらも、28名、つまり7割の人が「しかたない」とあきらめている。そして、勤務表作成を「やりたい」と思っているのは3名なのに対し、勤務表作成を「やりたくない」「できればやりたくない」と思っている人は、それぞれ6名と30名であり、全体の9割を占めていることもわかった。勤務表作成をやりたい理由としては、勤務表作成が病棟の運営において非常に重要なものであることが挙げられていた。一方、勤務表作成をやりたくない、できればやりたくない理由は、内容的に(1)勤務表作成に費やす時間に関するもの(2)勤務表作成の難しさを挙げたもの(3)ストレスを挙げたもの、の3つに分類できた。つまり、数多い条件の下での勤務表作成は非常に難しく、多くの作成時間を要し、プライベートの時間をも犠牲にしなくてはならないからである。しかし、この勤務表に対し、誰が作成するのが良いと思うかについては、8割の人が「主任」「婦長」を挙げていることから、「やりたくはないけれど自分がやらなくてはいけない仕事」と感じていることがわかる。

勤務表作成の手順

まず、勤務表をどのように作成しているかを知るために、作成に使用する「道具」と「情報」について整理した。道具は使用の目的によって、筆記用具(勤務表作成用紙1~6枚、鉛筆、記号3~11種類、消しゴム、定規、色鉛筆、色マジック、ボールペン、修正液)、資料(前月の勤務表、数ヵ月前の勤務表、有給帳)、計算道具(電卓、表計算ソフト)の3つに分けることができた(資料は以下の「情報」にも含まれる)。ここで、興味深いのは筆記用具の中のボールペンと定規である。この具体的利用法は、勤務表作成作業のタスク分析(嶋田 他、2001)で明らかになった。ボールペンで「決定済みの勤務や休み希望」を表すことにより、他の部分を鉛筆と消しゴムで何度も書いたり消したり試行錯誤を繰り返しながらの勤務表作成が可能になる。そして、長い定規を勤務表の縦1列にあて、その記号の並び(その日の各シフトのメンバー)を見たり、横1行にあて、その記号の並び(対象ナースのスケジュール)を見て検討、改善を重ねていく。これらの方法が、勤務表作成の考慮点に深く関わっていることが本調査の残りの部分で明らかになる。

情報については、変動のない個人情報、毎月変動のある個人情報、部署全体としての情報の3つに分けられたので、主なものを以下に示す。変動のない個人情報では(1)スキルに関するもの(現部署での年数、ナースとしての年数、シフトのリーダーを任せられるか否か、新人など教育的状況、年齢)、(2)規定されているもの(年休の付与日数、所属チーム)、(3)個人の都合(勤務の希望曜日、勤務シフトの希望順位、不可能な勤務シフト、通勤時間、習い事、育児、健康状態などからの制約)が挙げられた。変動のある個人情報では(1)業務予定(院内教育、カンファレンス、勉強会、ミーティングなど日常業務以外の予定)、(2)休み希望、勤務希望関係(休みの希望日、年休の残数、各勤務シフトの希望日、前月の勤務状況、希望受け入れ状況)、(3)個人の都合(健康状態がどうであるか、精神状態がどうであるか)等が上げられえた。勤務表作成に必要な全体情報では(1)以前の勤務表(前月の勤務表、4月からの勤務表)、(2)病院のスケジュール(院内教育、勉強会、カンファレンス、ミーティングなど日常業務以外の予定、オペや大きな検査の日程)、(3)メンバーの組合せ上の相性、等が挙げられた。

作成手順の大きな流れは、どの回答においても「優先順位の高い条件から順に、それらを満足するよう勤務を決定していく」というものであった。最優先する条件には、「行事」「個々の希望」「各シフトの人員確保」「各シフトのリーダーの確保」が挙げられていた。ここで行事とは、院内教育、勉強会、研修、会議等のことである。これら以外に優先する条件としては、「休みの数の均等化」「深夜勤回数の均等化」「準夜勤回数の均等化」「制約の強い人の都合」というものが2番目以降に挙げられていた。この他、「人数が足りない場合、それを補えるメンバー

にする」といったような「守るべき条件を満たせなかった場合の対応策」も挙げられていた。また、各日の勤務者数が不足しないように、勤務表作成前に「休み希望の出し方」について指示を出している場合が多い。休み希望が重なった場合には「各人で調整」させ、希望を出せる回数に制限や優先順位をつけさせたり、多くの人数を必要とする特定の日には希望を出させないといった「受け入れる希望を押える」ような指示を出している。

勤務表が満たすべき条件

勤務表作成において「絶対守るべき条件」に挙げられたものと「できれば守りたい条件」に挙げられたものをあわせると、どの部署においても、ほぼ同じような条件セットになっていた。これらの条件を、回答数が多かった順に表2に示す。

また、同じ項目が、両方に挙げられていることから、部署によって条件の重要度が異なることがわかる。内容としては、印をつけた項目が、各シフトの業務に支障を起こさないための「人員確保」と「メンバー構成」に関する条件、印をつけた項目が、各ナースの勤務負荷を考慮する「勤務や休みの回数」や「シフトパターン」、そして、各ナースの「希望する休み」を受け入れること等の条件となっている。

「人数の確保」や「勤務や休みの回数」については、その数の上下限を数値によって与える条件であるので、ここでは、それら以外の「メンバー構成条件」と「シフトパターン条件」が、具体的にどのように与えられているかを整理する。

ナース同士の組合せの条件については、ほとんどの部署(回答 39 部署中の 38 部署)で考慮されていた。その具体的内容は(1)スキル的に不安やかたよりのない組合せを考慮しているもの(リーダーをとれる人がいるように組合せる, 新人同士をさけて組合せる)(2)スキル以外のものでかたよりのない組合せを考慮しているもの(各チームのメンバーが均等に含まれる組合せ, 同じメンバーを何度も組合せない)(3)個人の組合せとして考慮しているもの(相性を考慮した組合せ, 慣れ合いのない組合せ)の大きく3つであった。

また、シフトパターン条件については、「好ましい基本パターン」があるとしている部署は半分以下だが、「禁止するパターンや好ましくないパターン」は、どの部署にも必ず存在する。しかし、基本パターンとして挙げられたものを観察すると、禁止するパターンや好ましくないパターンを考慮すれば、おのずと基本パターンが出来上がることがわかった。そこで、禁止するパターンや好ましくないパターンに注目すると、これらのパターンには、勤務が終了してから次の勤務までの時間が8時間を割るようなシフト間隔の少ないパターン、同じシフト(特に夜勤)が何日も続くパターン、休みが何日も入らないパターン、等が挙げられている。これらのシフトパターンを表3に示す。

ここで、シフトパターンについて補足しておく。一般に、3交替での1日は、深夜勤、日勤、

表2. 勤務表作成において守るべき条件。

(●:各シフトにおける人員確保とメンバー構成条件, ○:各ナースの勤務負荷を考慮する条件)

(1) 絶対守るべき条件	(2) できれば守りたい条件
○休みの回数	○希望する休み
●各シフトの人員確保	○準夜勤の回数
○深夜勤の回数	○深夜勤の回数
○準夜勤の回数	●各シフトの人員確保
○希望する休み	○休みの回数
●スキルからの組合せ	○シフトパターンについて
○シフトパターンについて	○公平さ
	●メンバーの組合せ

表 3. シフトパターンの例.

3 交替制	2 交替制
(1) 禁止パターン 準夜勤→日勤 深夜勤→準夜勤 日勤→深夜勤 深夜勤 2 連続以上 深夜勤 3 連続以上 準夜勤 3 連続以上 準夜勤 4 連続以上 勤務 7 連続以上	(1) 禁止パターン 夜勤→日勤 夜勤 2 連続以上 勤務 7 連続以上
(2) 好ましくないパターン 深夜勤→休み→日勤 準夜勤→日勤 深夜勤→深夜勤→休み→日勤 準夜勤 4 連続以上	(2) 好ましくないパターン 夜勤 2 連続以上 夜勤 3 連続以上 何日も日勤が続く

準夜勤, の順に構成されるのが標準的と言われているが, 東京女子医科大学附属病院では, 日勤, 準夜勤, 深夜勤の順で構成されている. 各シフトの時間帯は, 多少のバリエーションはあるが, 標準的には, 日勤: 8 時 ~ 16 時, 準夜勤: 15 時 30 分 ~ 23 時, 深夜勤: 22 時 30 分 ~ 翌朝 8 時 30 分であった. 例えば, パターン「深夜勤 準夜勤」は, 深夜勤の次の日が準夜勤である勤務を表し, 深夜勤の明けた同日の 15 時 30 分から準夜勤に入るものである. また, 2 交替制における夜勤は, 3 交替の準夜勤と深夜勤をあわせた時間帯のシフトであり, 勤務は 2 日間にわたる. よって「2 連続夜勤」というのは 4 日にわたる勤務であり, 2 回目の夜勤は 1 回目の夜勤入りの 2 日後に夜勤入りすることになる.

望ましい勤務表の条件

できあがった勤務表をみてどのような勤務表がよいと思うかという質問に対しては(1)個々の希望が達成されている(休み希望の達成, 勤務希望の達成, 勤務意欲をもてる, 納得できる), (2)公平さが保たれている(準夜勤回数, 深夜勤回数の均等化, 休み回数の均等化, 全体的な公平さ)(3)各シフトの人員確保とバランスよいメンバー構成(メンバー構成のバランス, 必要人数などの制約を守っている)(4)望ましいシフトパターンになっている(無理のないシフトパターン, シフト間隔など勤務のリズムがよいもの)といった内容のものが挙げられた.

ここで挙げられた条件は, 守るべき条件として挙げられた項目と, ほぼ一致していることがわかる.(3)は表 2 の 印を付けられた条件を満たしていること.(1)(2)(4)は 印を付けられた条件を満たしていることに対応している. これらの結果と「守るべき条件を全て満足する勤務表を作成することが難しい」という現実から, ナース勤務表作成においては, 守るべき条件をできる限り満たすことが目的になっていることがわかる.

スケジューリング・システムについての意見

勤務表作成をコンピュータでおこなうことを考える場合, 直接コンピュータに指示できないと思われている条件としては, 相性, 人間関係, 個々の能力, 性格, 体調, プライバシー, 人情といった, データとしてコンピュータ上に乗せにくいものの扱いが挙げられた. これは, 数値化不可能という意味と, 明確にすること自体に問題があるという意味の 2 つの解釈があると思われる. しかし, 勤務表作成者が勤務表を作成する時点においては, 具体的な「各シフトのメンバー構成」や「各ナースのシフトの並び」に対して, 好ましい, 好ましくない, 絶対許されない, 等と判断しているので, これらを直接明確にしないで扱える方法が存在すると思わ

れる。また、これら以外に挙げられていた中で「条件を満たせなかった場合の評価尺度」だけは人間の判断を完全に切り離すことが難しいものの「守るべき条件の扱い」はアルゴリズムでも対応可能であり、「勤務表の修正」はシステムのデザインで対応できるものとする。

また、勤務表作成を支援してくれるコンピュータ・システムがあったとしたら、どのようなものがよいか、4つの項目から選んでもらった結果は(1)与えた条件をチェックしてくれる表計算ワープロのようなもの(6名)(2)ある程度の条件を満たした勤務表を表示し、後で手直し可能なもの(24名)(3)与えた条件に対して自動で勤務表を与えてくれるもの(22名)(4)使いたくない(0名)、重複選択(8名)であった。ここで、「使いたくない」人がいないことに注目する。そして、38名までの人が(2)または(3)を選んでいたので、なんらかのスケジューリングの機能をもったシステムが望まれていることもわかった。

まとめ

アンケート調査以外にも、実際の勤務表を作成しながら作成担当者との議論を繰り返した。それらの議論からわかったことも合わせて、本調査と、その後の調査(23病院³¹⁵部署を対象とした調査)で明らかになった勤務表作成における考慮点を以下に整理しておく。

各日の各シフトにどのメンバーを配置するかを決定するための条件は、ナースの能力レベルを3段階前後(高々6段階)に分け「勤務や能力で分けたグループからの人数に制限をつける」というものである。例えば「深夜勤には新人は1人以下になるように」「リーダーレベルのナースが1人以上入るように」というように考えている。これは、回答のあったほとんど全ての部署において共通のことであった。また、教育面、そして業務量や業務の内容により配置人数やメンバー構成レベルを変えたり、人間関係により特定のナース同士の組合せを避けたりしている。

一方、各ナースのスケジュールに関する条件としては、「期間内の休みや各シフトの回数」や「土日祭日にあたる休みのあげ方」そして「休みや勤務の希望」や「研修会や他部署との関連等の日常業務以外の勤務」を考慮する以外に、「夜勤を3回以上続けない」といったような同一シフト連続数の制限「夜勤と夜勤の間は4日以上あける」といったような同一シフトの間隔に対する制限「夜勤の次の日に日勤を入れない」といったような異種シフトの並びに対する制限、等を考えていることがわかった。また、全てのナース共通の条件の他に、ナース毎に、業務スケジュール、休みや勤務希望、体調等を考慮しなければいけないことがわかった。

また、現実には、全てを満たす勤務表の作成が困難であることから、勤務表作成担当者は、これらの条件をできる限り満たすことを目標に勤務表を作成している。

一方、どうしても全ての条件を満たす勤務表ができない場合には、個々のナースのスケジュールに関する条件をあきらめる(一部のナースに我慢してもらう)ことで対応することが多い。休み希望・勤務希望を受け入れないことや、休みの間隔、夜勤の間隔、夜勤連続数、他、シフトの並びを厳しくしたり、休みやシフトの数の公平さをあきらめるが、これらについては数ヶ月の単位で公平になるように対応している。また、シフトに関する条件の中では、日勤についての条件が比較的厳しくない。日勤には、ナース・スケジュールリングの対象となるローテーション・ナースの他に婦長や主任が勤務しているので、人数を減らしたりメンバーの組合せレベルを下げて、欠けた要素を婦長や主任が補えるからである。逆に、夜勤(または深夜勤や準夜勤)における人数やメンバー構成についての条件は、必ず守らなくてはならない。

4. モデル

我々の2つの調査の結果からは、ナース・スケジュールリングで考慮する全ての条件は、本質的に異なる2つのタイプのグループ;各シフトのスキルレベルを守るもの(シフト拘束条件)と、各ナースの負荷に関わるもの(ナース拘束条件)とに分けられることがわかった。このように拘

束条件を分けて考えることは、ある拘束条件が満たされない場合、なにが影響を受けてしまうのかを意識できる利点がある。

この章では、これらの拘束条件の詳細とこの問題の目的に関する考え方を述べたあと、この拘束条件のグループ化に基づく定式化を紹介する。

シフト拘束条件

シフト拘束条件は、各シフトに適した人数とスキルレベルのナースを割り当てることにより看護の質を守ろうというものである。具体的には、各シフトの合計勤務人数や各グループからの人数に下限値と上限値を設定するものである。

相性の悪い(もしくは馴れ合いのある)ナースの組合せを避けたり、夜勤における同じ組合せの繰り返しを避ける場合にも、対象となるナースだけのグループ(相性の悪いナース 2 名のみのグループや先月夜勤で組んだナース同士のグループ)を考慮対象分だけ設定し、グループからの対象シフト(例えば夜勤)への人数の上限値を 1 と設定することで考慮できる。

ナース拘束条件

ナース拘束条件は、以下の 3 つのタイプの拘束条件からなり、各ナースの労働負荷を考慮するものである。Type I: 各シフト(日勤/夜勤/休み、または、日勤/準夜勤/深夜勤/休み)が適切な回数であること、Type II: 休みやシフトの希望日やセミナー参加を達成すること、Type III: ナースの健康に悪影響をもたらすシフトの並びを避けること。

Type I は、その数に上下限値を設定することで簡単に定式化することができる。Type II もナースの希望に関係する決定変数を固定することで簡単に定式化することができる。しかし、Type III は、さらに以下の 3 つに分類できるものの、その定式化は難しい。(i)同一シフトの連続回数に関するもの。例えば「3 連続夜勤は許されない」等 (ii)同一シフトの間隔に関するもの。例えば「夜勤と夜勤の間は少なくとも 4 日はあける」「休みと休みの間は 6 日以下」等。(iii)異種シフトの並びに関するもの。例えば「夜勤翌日の日勤は許されない」等。

目的

この問題の目的は、なんらかのコストを最小化したいというより、以上で述べた拘束条件を全て満たすことである。従って、ここでは、これら全ての要望を問題の拘束条件として表現し、目的関数を意識しなくても良いよう融通のきく形で表現することにする。

上記に挙げたシフト拘束条件とナース拘束条件以外に守るべき条件や目標が存在する場合にも、これら 2 つのバリエーションとして扱えることが、3 章で述べた調査からわかっている。例えば、全てのナースに週末 2 連休を与えたいという目標は、Type I や Type II の拡張と考え、ナース拘束条件として扱うことができる。

意思決定変数

1 人のナースの実行可能シフトパターンを「そのナースに関する全てのナース拘束条件を満たすスケジュール」と定義する。この実行可能シフトパターンを各ナースに対して陽的もしくは陰的に用意しておいた下で、シフト拘束条件を満たすだけでなく目的関数をも最小化するような「実行可能シフトパターンの組合せ」を見つけるものとして、この問題を定式化する。そして、ナース i が実行可能シフトパターン q を割当てられるときに 1、そうでないときに 0 となる意思決定変数 λ_{iq} を使うことにする。

記号

$M = \{1, 2, \dots, m\}$: ナース番号の集合。

$N = \{1, 2, \dots, n\}$: スケジューリング対象日の集合。

$W = \{\text{shift}1, \text{shift}2, \dots, \text{shift } w\}$: シフト(例えば, 日勤, 夜勤, 休み)の集合 .

$R = \{r | r \text{ はグループ}\}$: スキルや担当患者で分けられたグループの集合 .

$G_r = \{i | \text{ナース } i \text{ はグループ } r \text{ に所属}\}, r \in R$.

$a_{rjk}, r \in R, j \in N, k \in W$: j 日のシフト k に対するグループ r からの人数の下限值 .

$b_{rjk}, r \in R, j \in N, k \in W$: j 日のシフト k に対するグループ r からの人数の上限値 .

P_i : ナース i の n 日分の実行可能シフトパターン(スケジュール)の集合 . 各パターンは δ_{iqjk} (パターン q の j 日にシフト k が割当てられていれば 1, そうでなければ 0) で表現されている .

$S = \{s | s \text{ は 1 つの目標}\}$: 目標の集合 .

$f_s(\lambda_{iq}, i \in M, q \in P_i), s \in S$: 目標 s が満たされない度合いを表す関数 .

定式化 1 . 実行可能シフトパターンの最適な組合せを見つける問題として定式化する . λ_{iq} は, 各ナースに採用されたパターンを表現する意思決定変数である .

$$(4.1) \quad \text{Minimize } \sum_{s \in S} f_s(\lambda_{iq}, i \in M, q \in P_i)$$

subject to

$$(4.2) \quad a_{rjk} \leq \sum_{i \in G_r} \sum_{q \in P_i} \delta_{iqjk} \lambda_{iq} \leq b_{rjk} \quad r \in R, j \in N, k \in W$$

$$(4.3) \quad \sum_{q \in P_i} \lambda_{iq} = 1 \quad i \in M$$

$$(4.4) \quad \lambda_{iq} = 0 \text{ or } 1 \quad i \in M, q \in P_i$$

この定式化では, ナース拘束条件を全て満たした実行可能なスケジュールを使っているので, シフト拘束条件だけが目に見える形で表現されている . この論文では, 定式化 1 をベースに議論を進めるが, ナース・スケジューリング問題の入力パラメタが何であるのかを全てを明らかにするために, 両方のタイプの拘束条件を陽的に表現できる, もう 1 つの定式化を以下に示しておく(この論文では, この定式化 2 を 7 章の計算実験でのみ利用する) .

定式化 2 . 各ナースの各日各シフトに対し勤務とするか否かを決定することで, 最適な組合せを見つける問題として定式化する . x_{ijk} は各ナースの各日の勤務シフトを表現する意思決定変数である .

$$(4.5) \quad \text{Minimize } \sum_{s \in S} f'_s(x_{ijk}, i \in M, j \in N, k \in W)$$

subject to

$$(4.6) \quad a_{rjk} \leq \sum_{i \in G_r} x_{ijk} \leq b_{rjk} \quad r \in R, j \in N, k \in W$$

$$(4.7) \quad c_{ik} \leq \sum_{j \in N} x_{ijk} \leq d_{ik} \quad i \in M, k \in W$$

$$(4.8) \quad x_{ijk} = \tau \quad (i, j, k) \in F_\tau, \tau \in \{0, 1\}$$

$$(4.9) \quad \sum_{\alpha=1}^h x_{i \cdot j + \alpha - 1 \cdot k_\alpha} \leq h - 1 \quad i \in M, j \in \{1, \dots, n - h + 1\},$$

$$(k_1, k_2, \dots, k_h) \in P_h, h \in \{2, 3, \dots\}$$

$$(4.10) \quad u \leq \sum_{\alpha=1}^h x_{i \cdot j + \alpha - 1 \cdot k} \leq v \quad i \in M, j \in \{1, \dots, n - h + 1\},$$

$$(k, u, v) \in Q_h, h \in \{2, 3, \dots\}$$

$$(4.11) \quad \sum_{k \in W} x_{ijk} = 1 \quad i \in M, j \in N$$

$$(4.12) \quad x_{ijk} = 0 \text{ or } 1 \quad i \in M, j \in N, k \in W$$

ここで, $F_1 = \{(i, j, k) \mid \text{ナース } i \text{ の } j \text{ 日はシフト } k \text{ に固定}, i \in M, j \in N, k \in W\}$, $F_0 = \{(i, j, k) \mid \text{ナース } i \text{ の } j \text{ 日のシフト } k \text{ は不可}, i \in M, j \in N, k \in W\}$, $P_h = \{(k_1, k_2, \dots, k_h) \mid \text{シフト } k_1, k_2, \dots, k_h \text{ の連続勤務は禁止}, k_1, k_2, \dots, k_h \in W, h \in \{2, 3, \dots\}\}$, $Q_h = \{(k, u, v) \mid h \text{ 日間にシフト } k \text{ の回数は } u \text{ 以上 } v \text{ 以下}, k \in W, u, v \in \{0, 1, 2, \dots\}, h \in \{2, 3, \dots\}\}$, $c_{ik}, i \in M, k \in W$: ナース i のシフト k の回数の下限值, $d_{ik}, i \in M, k \in W$: ナース i のシフト k の回数の上限値, $x_{ijk}, i \in M, j \in N, k \in W$: ナース i が j 日のシフト k に勤務するとき 1, そうでないとき 0 となる意思決定変数, $f'_s(x_{ijk}, i \in M, j \in N, k \in W), s \in S$: 目標 s が満たされない度合いを表す関数とする。

5. 部分問題軸アプローチ

前章で明らかにした定式化 1 では, シフト拘束条件よりナース拘束条件に高い優先度が与えられている. この章では, この定式化の利点を述べ, この問題の構造をうまく利用したアプローチを提案する.

ナース・スケジューリングの扱いにくさの 1 つに, 所属するナースの人数やスキルレベルにも関わらず(例えば, 不足気味にも関わらず), 勤務表作成者が「実行不可能になってしまうほど多くの拘束条件を課してしまう」ことが挙げられる. 実際, 全ての拘束条件を満たす勤務表を作成できないことがしばしば起こるといわれ, 一般的にはナース拘束条件を緩和することで対応するという. しかし, この緩和をおこなえるのは, スタッフ・ナースの好み, 性格, 健康状態等をよく把握している人間だけ(師長や主任)である. よって, ナース・スケジューリングのためのアルゴリズムは, それらの条件を安易に扱うべきではなく, 勤務表作成者が後から手直ししやすいように, 十分な柔軟性を持った勤務表を与えるべきであると考えられる.

3 交替の問題にシミュレーテッド・アニーリングを適用した我々の論文, 池上 他(1996)では, 各シフトに必要なナースの合計数だけ満たす初期解を作成し, ある日のある 2 人のナースの勤務シフトを交換したものを近傍として設定した. 実際の勤務表を作成した結果は, ほとんどの拘束条件を満たすことができたものの, Type III のナース拘束条件を違反する部分をどうしても取り除くことができず, その拘束条件に高いペナルティコストを課してすらも不可能であった.

しかし, ナース拘束条件をいくつか満たしていない勤務表を勤務表作成者に手直ししてもらったときに, シフト拘束条件を満たしたまま手直しすることは非常に難しいということがわかった. そして, いくつかのシフト拘束条件を満たしていない勤務表に対して, ナース拘束条件の一部を適切に緩和しながら, 実際に使える勤務表に手直しする方がはるかに簡単であることもわかった. これは, ナース拘束条件には多くのタイプのものが存在し, その条件の重要度はナース毎に異なるうえ, そのナースのその他の日のスケジュールによっても異なってくるか

らである．よって，勤務表作成者にナース拘束条件緩和の自由度を与えるためには，いくつかのシフト拘束条件を満たすことに失敗してもナース拘束条件を全て満たすことができるアルゴリズムを構築する必要があった．そして，ナース拘束条件をシフト拘束条件と別扱いにして解くことが，実用上有効であると考えた．

問題分解の観点から見ると，ナース・スケジューリング問題の拘束条件は，ブロック対角構造をしている(Chvatal, 1983)．この構造の中では，各ナースの拘束条件が対角に並んだ各ブロックに対応し，それらは結合制約を無視すれば独立に扱うことができる．シフト拘束条件は結合制約に対応するが，ナースの人数の上下限条件なので扱いやすい．ブロック内のナース拘束条件の合計数に比べて，はるかに数が少ない．

定式化

ここで，前章の問題(4.1)-(4.4)の実行可能性をゴールとする定式化を考える．これは， λ_{iq} の値の変更，つまりナース拘束条件を満たしている実行可能パターンを入れ替えることにより，シフト拘束条件に関して違反する度合いを最小化しようというものである．

$$(5.1) \quad \text{Minimize} \quad \sum_{\lambda_{iq}, i \in M, q \in P_i} \sum_{r \in G} \sum_{j \in N} \sum_{k \in W} w_{rjk}^- \alpha_{rjk}^- + \sum_{r \in G} \sum_{j \in N} \sum_{k \in W} w_{rjk}^+ \beta_{rjk}^+$$

subject to

$$(5.2) \quad \sum_{i \in G_r} \sum_{q \in P_i} \delta_{iqjk} \lambda_{iq} + \alpha_{rjk}^- - \alpha_{rjk}^+ = a_{rjk} \quad r \in R, j \in N, k \in W$$

$$(5.3) \quad \sum_{i \in G_r} \sum_{q \in P_i} \delta_{iqjk} \lambda_{iq} + \beta_{rjk}^- - \beta_{rjk}^+ = b_{rjk} \quad r \in R, j \in N, k \in W$$

$$(5.4) \quad \sum_{q \in P_i} \lambda_{iq} = 1 \quad i \in M$$

$$(5.5) \quad \lambda_{iq} = 0 \text{ or } 1 \quad i \in M, q \in P_i$$

$$(5.6) \quad \alpha_{rjk}^-, \alpha_{rjk}^+, \beta_{rjk}^-, \beta_{rjk}^+ \geq 0 \quad r \in R, j \in N, k \in W$$

ここで， $w_{rjk}^-, w_{rjk}^+ \geq 0$ は，シフト拘束条件の重み付けであり， $\alpha_{rjk}^-, \alpha_{rjk}^+, \beta_{rjk}^-, \beta_{rjk}^+ \geq 0$ は，スラックまたはサープラス変数である．通常， $w_{rjk}^-, w_{rjk}^+ = 1$ と設定することにするが，これは，これまでの経験で，このシンプルな設定が有効だったからであり， w_{rjk}^-, w_{rjk}^+ に対する他の重み付けの利点を見つけられなかったからである．

しかしながら，これらの重み付けをあえて表現することにより，変動ナース(floating nurse)のコストを最小化するタイプのナース・スケジューリング問題に対しても，「 j 日のシフト k に変動ナースを利用する場合にかかるコスト」を w_{rjk}^- に直接代入することで，適用できるようになっている．これは，国外のナース・スケジューリングへの幅広い適用可能性を示す重要なポイントである．

部分問題と部分問題軸アプローチ

ナース $i \in M$ を選んで，それ以外のナースのスケジュールを与えられた試行解のもので固定した場合，定式化の(5.1)-(5.6)は，このブロック対角構造問題の部分問題と等しくなっており，

その目的関数は結合制約を満たそうというものである．ナース i についての部分問題は，他のナースのスケジュールを固定して目的関数を最小化する問題であり，以下のように表される：

$$(5.7) \quad \text{Minimize } \sum_{\lambda_{i,q}, q \in P_i} \sum_{r \in G} \sum_{j \in N} \sum_{k \in W} w_{rjk}^- \alpha_{rjk}^- + \sum_{r \in G} \sum_{j \in N} \sum_{k \in W} w_{rjk}^+ \beta_{rjk}^+$$

subject to

$$(5.8) \quad \sum_{i' \in G_r} \sum_{q \in P_{i'}} \delta_{i'qjk} \lambda_{i'q} + \alpha_{rjk}^- - \alpha_{rjk}^+ = a_{rjk} \quad r \in R, j \in N, k \in W$$

$$(5.9) \quad \sum_{i' \in G_r} \sum_{q \in P_{i'}} \delta_{i'qjk} \lambda_{i'q} + \beta_{rjk}^- - \beta_{rjk}^+ = b_{rjk} \quad r \in R, j \in N, k \in W$$

$$(5.10) \quad \sum_{q \in P_i} \lambda_{iq} = 1$$

$$(5.11) \quad \lambda_{iq} = 0 \text{ or } 1 \quad q \in P_i$$

$$(5.12) \quad \alpha_{rjk}^-, \alpha_{rjk}^+, \beta_{rjk}^-, \beta_{rjk}^+ \geq 0 \quad r \in R, j \in N, k \in W$$

ここで，この構造をうまく利用するために，以下のような反復方策を使うことにする．適当な試行解からスタートし，各ナースについて部分問題(5.7)-(5.12)を解き，それらの結果の中から最良な解を次の試行解として採用する．そして，このプロセスを目的関数値が 0 になるまで繰り返す．

このアプローチは，問題分解に基づく局所探索の 1 つである．個々のナースに関わる部分問題に焦点を当てながら全体問題を扱うことから，線形計画法における分解アルゴリズム(decomposition algorithm)との混同を避けるため，部分問題軸アプローチ(Subproblem-centric approach)と呼ぶことにする．

さらに，もっと広い意味で部分問題軸アプローチを捉えて，目的関数に「結合制約を違反する度合い最小化」を含むような部分問題を，繰り返し解くことによって目的関数を最小化する「ブロック対角構造問題のためのアプローチ」として考えることにする．

6. 2 交替用アルゴリズム

部分問題軸アプローチの有効性を評価するために，我々は最初に 2 交替のためのアルゴリズムを構築した．

2 交替における夜勤に関する拘束条件は，日勤のものに比べて非常に厳しくきつく与えられている．これは，夜勤の時間(通常午後 4 時から翌朝 8 時まで)が非常に長く，夜勤に関する負荷を十分考慮する必要があるからである．さらに，夜勤のナース数は少ないうえ師長や主任が不在であることから，スキルレベルを維持することも難しくなっている．

このような観点から，この問題を 2 フェーズで解くことにする．夜勤の拘束条件の厳しい性質を考慮し，アルゴリズムでは，それらを日勤の拘束条件よりはるかに重視して初めに扱う(フェーズ I)．フェーズ II では，フェーズ I で得られたスケジュールを基に，日勤の拘束条件を満たすスケジュールを見つける．

2 フェーズに分けて解くことは、各フェーズにおける実行可能シフトパターンの全列挙を可能にし、フェーズ I でもフェーズ II でも、部分問題はこの列挙に基づいて簡単に解くことができるようになる。

初めに、フェーズ I で解かれる「2 交替問題の緩和問題」を定義する。この緩和問題では、ある日勤 (day と表す) で働くナースの数に対する上限値 $b_{rj\text{-day}}$ を緩和する。さらに、各ナース i に対して、日勤の数に対する拘束条件、休みに対する拘束条件、日勤と休みに関する並びの条件を緩和する。フェーズ I で解かれる緩和問題を以下に示す。

緩和問題・問題(5.1)(5.6)において、日勤の数の上限値条件に対する重み付けを、

$$(6.1) \quad w_{rj\text{-day}}^+ = 0 \quad r \in R, j \in N$$

と設定し、日勤と休みに関する Type I と III のナース拘束条件を緩和した下で、ナース $i \in M$ の P_i を設定する。

この緩和問題を解くための、ナース $i \in M$ に対する P_i の作成は以下の通りである。「日勤と休みに対する Type I と III 以外のナース拘束条件」を満たしながら、できる限り多くの日勤が含まれるようにパターン $q \in P_i$ を作成する。表 4 は、「 $4 \leq$ 夜勤の数 ≤ 5 」「夜勤と夜勤の間の日数 ≥ 3 」「14 日にセミナーを入れる」「週末の 2 連休を確保する」を満たすパターン $q \in P_i$ の例を示す。さらに、パターン作成では、前月末のスケジュールからのシフトの並びについても考慮しなければならない。

緩和問題の実行可能シフトパターンは、元の問題の実行可能シフトパターンを陰的に表現している。実際、このパターンのいくつかの日勤を休みに入れ替えることで元の問題の実行可能パターンを作成することができる。この緩和を利用することにより、各ナースの実行可能シフトパターンを実際よりかなり少ない数だけで評価できるので、部分問題が簡単に解けるようになっていく。そこで、この緩和問題に部分問題軸アプローチを適用する(フェーズ I)。

緩和問題を解いた結果は、各ナースのスケジュールが日勤過剰になっているので、フェーズ II では、いくつかの日勤を休みに入れ替えることで、その数を削減する。しかし、グループ r のナースが(フェーズ I で)日勤に割当てられた数と下限値 $a_{rj\text{-day}}$ が等しくなった場合については、それらのナースの日勤をあらかじめ固定することを忘れてはならない。よって、フェーズ I で得られたナース i のシフトパターンに対して、固定されていない日勤のいくつかを休みに入れ替えることにより、元々のナース拘束条件の全てを満足する実行可能シフトパターンを作成し、新しい P_i に設定する。新しい $P_i, i \in M$ に対して、再び部分問題軸アプローチを適用するが、ここでは、全ての $\alpha_{ij\text{-night}}^-, \beta_{ij\text{-night}}^+, r \in R, j \in N$ がすでにゼロに固定されているので、 $w_{ij\text{-night}}^-, w_{ij\text{-night}}^+, r \in R, j \in N$ は、意味を持たない(ここで、'night' は、2 日にわたる夜勤の 1 日目を表すものとする)。

それぞれのフェーズでは、重み付けの値 $w_{rj\text{-day}}^+$ や実行可能シフトパターンが異なるものの、アルゴリズムは同じフレームで実行できる。各フェーズの最初の試行解には、各ナースに対し

表 4. フェーズ I での実行可能パターン(Nn:夜勤, -:日勤, +:セミナー / :休み)。

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S
-	/	/	N	n	/	-	-	N	n	/	-	-	+	N	n	/	-	-	N	n	/	-	-	N	n	/	-	-	-

週末 2 連休

少なくとも 3 日間

て全ての日が休みであるようなダミー・スケジュール q_0 を利用する (δ_{iqjk} は、パターン q の j 日にシフト k が割当てられる場合に 1, そうでない場合に 0, と定義されている).

$$(6.2) \quad q_0 : \text{if } k = \text{off } \delta_{iq_0jk} = 1, \text{ otherwise } \delta_{iq_0jk} = 0, j \in N$$

アルゴリズムの速度向上のため、フェーズ I におけるダミー・パターンが取り除かれた後では、現在解との差が小さいスケジュールだけ調べたことを考えた。ここで、差を「一方の解で夜勤になっているが、もう一方の解では夜勤になっていない日」の数と定義する。アルゴリズムが対象を絞込むために使用する差を $DIFF$ で表すことにする。

q^i を現在の試行解で採用されているナース i のパターンとする。そして、 \tilde{P}_i を以下のように定義する：もし $q^i = q_0$ ならば、 $\tilde{P}_i = P_i$, そうでないならば、

$$(6.3) \quad \tilde{P}_i = \left\{ q \mid \sum_{j=1}^n |\delta_{iqj \cdot \text{night}} - \delta_{iq^i j \cdot \text{night}}| \leq DIFF, q \in P_i \right\}.$$

さらに、無限ループを避けるために、試行解に採用されていたスケジュールをしばらくの間、比較プロセスから外すことを考える。各イテレーションで、新しい解が試行解として採用されたときに取り除かれるパターンを憶えておくために、tabu リストを利用する。よって、アルゴリズムの各フェーズは tabu 探索をおこなっていることになる。

TL を tabu リストの長さとする。アルゴリズムは、現在の試行解を更新する際、直前の TL イテレーションで取り除かれたパターンを無視することになる。そこで、 $removed_{iq}$ を、パターン q が試行解から取り除かれたときのイテレーションの番号とする。そして、 z_i を部分問題 i の最適目的関数値、 i^* を $z_{i^*} = \min_{i \in M} z_i$ を満たすものとする。ことにより、アルゴリズムは、次の手順で表すことができる。

アルゴリズム

フェーズ I

- Step 1. 各ナース $i \in M$ に対し、緩和問題のナース拘束条件の下で実行可能シフトパターン集合 P_i を作成。各 $q \in P_i$ に対し、 $removed_{iq} = -\infty$ 。
- Step 2. ダミーパターン q_0 を作成。各ナース $i \in M$ に対し、 $P_i = P_i \cup \{q_0\}$ 。
- Step 3. 各ナース $i \in M$ に対し、 $q^i = q_0$ 。
- Step 4. 各ナース $i \in M$ に対し、 $\tilde{P}_i = P_i$ 。iteration = 1。
- Step 5. 各ナース $i \in M$ に対し、 $TABU_i = \{q \mid removed_{iq} \geq iteration - TL, q \in P_i\} \cup \{q^i\}$ 。 q^i と交換した場合に目的関数最小となるパターン $q \in (\tilde{P}_i \setminus TABU_i)$ を見つけ、この最良パターンを q_i 、その目的関数値を z_i とする。
- Step 6. $z_{i^*} = \min_{i \in M} z_i$ となる i^* を選び、次の試行解用に q_{i^*} を採用：
 $q' = q^{i^*}$, $q^{i^*} = q_{i^*}$ 。 $q' = q_0$ ならば、 $P_{i^*} = P_{i^*} \setminus \{q_0\}$ 。
 $\tilde{P}_{i^*} = \{q \mid \sum_{j=1}^n |\delta_{iq^i j \cdot \text{night}} - \delta_{iq^{i^*} j \cdot \text{night}}| \leq DIFF, q \in P_{i^*}\}$ 。
 $removed_{i^* q'} = iteration$ 。 iteration = iteration + 1。
- Step 7. $z_{i^*} = 0$ ならば、現在の $q^i, i \in M$ を緩和問題の実行可能解として、Step 8 へ。そうでない場合、Step 5 へ。

フェーズ II

- Step 8. 各ナース $i \in M$ に対し、 q^i に基づく実行可能シフトパターンを作成し、新しい P_i を設定。各 $q \in P_i$ に対し、 $removed_{iq} = -\infty$ 。

- Step 9. ダミーパターン q_0 を作成 . 各ナース $i \in M$ に対し , $P_i = P_i \cup \{q_0\}$.
- Step 10. 各ナース $i \in M$ に対し , $q^i = q_0$.
- Step 11. $iteration = 1$.
- Step 12. 各ナース $i \in M$ に対し , $TABU_i = \{q | removed_{iq} \geq iteration - TL, q \in P_i\} \cup \{q^i\}$. q^i と交換した場合に目的関数最小となるパターン $q \in (P_i \setminus TABU_i)$ を見つけ , この最良パターンを q_i , その目的関数値を z_i とする .
- Step 13. $z_{i^*} = \min_{i \in M} z_i$ となる i^* を選び , 次の試行解用に q_{i^*} を採用 :
 $q' = q^{i^*}$, $q^{i^*} = q_{i^*}$. $q' = q_0$ ならば , $P_{i^*} = P_{i^*} \setminus \{q_0\}$.
 $removed_{i^*q'} = iteration$. $iteration = iteration + 1$.
- Step 14. $z_{i^*} = 0$ ならば , 終了 . そうでない場合 , Step 12 へ .

ここで記述はしなかったが , 実行可能解が得られないときのために , 終了ルールが必要である . 実際 , 各イテレーションの終了条件として , イテレーションの数に上限値を設定している . TL のサイズについても , 扱う問題に従いチューニングがなされる . さらに , 勤務表作成者自身が各フェーズで解を選べるように , アルゴリズムは , 最小の z の値を持つ解を複数得る必要がある .

アルゴリズムが現実のデータ Ikegami-2shift-DATA1 に対して全ての拘束条件を満たすスケジュールを得た例を以下に示す . $TL = 30$, $DIFF = 3$ と設定した場合 , アルゴリズムはフェーズ I のイテレーション 47 において , 表 5 に示す解 (空欄はまだ日勤か休みが決定していないことを表す)を得た後 , フェーズ II のイテレーション 31 において , 2 章の表 1 で示した解 (スケ

表 5. フェーズ I の結果のスケジュール (2shift-DATA1) .

(-:日勤, Nn:夜勤, +:その他の勤務, /:休み)

Nurse	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	/	-	Nn	+	
No.	F	S	S	H	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	/	-	Nn	+	
1	n	/	/	-	N	n	/	-	-	-	+	N	n	/	-	-	N	n	/	-	-	N	n	/	/	/	N	n	/	8	10	5	1		
2	/	N	n	/	-	N	n	/	/	-	N	n	/	-	-	N	n	/	-	-	N	n	/	-	-	/	/	-	-	9	11	5	0		
3	N	n	/	/	N	n	/	-	N	n	/	-	-	/	-	-	-	-	-	-	-	-	-	N	n	/	-	-	N	6	15	5	0		
4	/	/	/	/	-	-	N	n	/	-	-	N	n	/	-	-	-	-	/	N	n	/	-	-	N	n	/	-	-	8	14	4	0		
5	-	N	n	/	-	-	N	n	/	-	-	-	N	n	/	-	-	-	N	n	/	/	/	N	n	/	-	-	-	7	13	5	0		
6	/	-	N	n	/	-	-	N	n	/	-	-	-	N	n	/	-	-	-	-	-	-	/	/	-	-	N	n	/	7	15	4	0		
7	-	-	-	/	-	+	N	n	/	-	-	N	n	/	/	/	-	-	N	n	/	-	-	-	-	-	N	n	/	7	14	4	1		
8	N	n	/	-	+	+	N	n	/	-	-	N	n	/	-	-	-	-	N	n	/	/	/	N	n	/	-	-	-	7	11	5	2		
9	/	/	-	N	n	/	-	-	-	+	-	-	+	-	-	N	n	/	-	-	-	-	N	n	/	/	N	n	/	7	14	4	1		
10	-	-	N	n	/	/	-	-	N	n	/	-	-	-	-	/	/	N	n	/	-	-	-	-	-	-	-	N	n	6	16	4	0		
11	/	-	N	n	/	-	-	-	N	n	/	+	-	-	N	n	/	-	-	-	N	n	/	/	/	/	/	-	-	N	n	9	10	5	1
12	N	n	/	/	-	-	-	-	N	n	/	-	-	-	N	n	/	-	-	-	-	-	-	-	-	N	n	/	-	-	5	17	4	0	
13	/	N	n	/	-	-	N	n	/	/	-	-	-	N	n	/	-	-	-	-	-	-	-	-	-	-	-	-	-	7	13	5	0		
14	-	-	N	n	/	-	-	-	-	-	-	-	/	/	/	/	N	n	/	-	-	-	-	-	-	-	-	-	N	7	16	4	0		
15	n	/	/	/	-	-	+	N	n	/	-	-	-	-	N	n	/	-	-	-	/	N	n	/	-	-	-	N	n	/	8	12	4	1	
16	N	n	/	-	-	N	n	/	-	-	N	n	/	-	-	/	/	-	-	-	-	-	-	-	-	-	-	-	-	7	13	5	0		
17	-	-	-	N	n	/	-	-	-	-	+	N	n	/	-	-	-	-	-	N	n	/	-	-	-	-	-	-	-	6	15	4	1		
18	/	/	-	-	N	n	/	-	-	-	+	-	-	+	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	6	15	4	1		
19	-	/	/	-	-	-	-	-	/	/	+	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	3	25	0	2		
20	-	-	N	n	/	/	-	N	n	/	-	-	-	-	N	n	/	-	-	-	-	-	-	-	-	-	-	-	-	7	13	5	0		
21	-	-	N	n	/	-	-	N	n	/	-	-	-	-	N	n	/	-	-	-	-	-	-	-	-	-	-	-	-	6	14	5	0		
22	N	n	/	-	-	-	N	n	/	/	-	-	-	-	N	n	/	/	-	-	-	-	-	-	-	-	-	-	-	7	13	5	0		
23	N	n	/	/	-	+	-	-	N	n	/	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	6	13	5	1		
24	-	-	N	n	/	+	/	/	N	n	/	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	6	15	4	1		
25	n	/	/	-	-	N	n	/	-	-	-	-	-	-	N	n	/	-	-	-	-	-	-	-	-	-	-	-	-	6	14	5	0		
26	/	-	-	N	n	+	/	-	-	-	N	n	/	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	7	14	4	1		
27	n	/	/	-	-	-	N	n	/	-	+	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	8	11	4	2		
28	-	-	-	-	-	-	-	-	/	/	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	4	21	3	0		
-	13	15	9	10	15	14	11	14	12	9	15	10	16	12	15	11	9	16	16	13	15	16	11	10	15	15	13	15	16	16					
Nn	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4					

表 6. フェーズ I における各ナースの実行可能パターン(2shift-DATA1).

A team										
Nurse No.	1	2	3	4	5	6	7	8	9	10
Number	1965	2752	1925	4222	9164	6187	1906	710	782	6502
B team										
Nurse No.	11	12	13	14	15	16	17	18	19	—
Number	1949	10173	1200	4269	134	235	3951	4614	5	—
C team										
Nurse No.	20	21	22	23	24	25	26	27	28	—
Number	9260	13150	6171	5420	3767	177	519	354	6601	—

表 7. フェーズ II における各ナースの実行可能パターン(2shift-DATA1).

A team										
Nurse No.	1	2	3	4	5	6	7	8	9	10
Number	8	10	229	15	109	224	6	120	49	47
B team										
Nurse No.	11	12	13	14	15	16	17	18	19	—
Number	9	2168	120	12	6	94	94	157	225	—
C team										
Nurse No.	20	21	22	23	24	25	26	27	28	—
Number	109	149	136	136	121	56	56	55	486	—

ジュール)を得た。実行時間は、GATEWAY G6-226(Pentium II, 266MHz, FreeBSD2.2.2)上で、それぞれ 11.5 秒と 1.4 秒であった。表 6, 表 7 に、フェーズ I, フェーズ II における実行可能パターンの数(P_i に含まれるパターンの数)をそれぞれ示す。

適切な値の $DIFF$ (例えば, 2 または 3) を設定したアルゴリズムは、ここでの問題例より数も多くきつい拘束条件を持つ問題に対しても、実際の問題を解くのに十分な速さで実行可能解を得ている(池上, 2000)。

7. 2 交替にも 3 交替にも可能なアルゴリズム

前章のアルゴリズムは、列挙に基づき、2 交替のナース・スケジューリング問題の実際の問題例を解くことに成功した。しかしながら、同じ方策をもっと複雑な 3 交替に拡張しようとすると、3 交替には 2 種類の夜勤(準夜勤, 深夜勤)があるために困難となってしまう。この章では、2 交替も 3 交替も同様に解けるようなアルゴリズムの構築を試みる。部分問題軸アプローチに基づくシンプルなアルゴリズムを使って、それが良い解を得ることができるのか確かめるためにおこなった予備実験の結果を紹介する。

当初、前章のアプローチを直接拡張し、問題を 3 フェーズで扱うアルゴリズムを構築した。このアルゴリズムは、フェーズ I で日勤と準夜勤の上限を緩和し、フェーズ II で日勤の上限を緩和すると同時に、Type I と III のナース拘束条件の一部を緩和して、それぞれの P_i を作成した。よって、フェーズ III は、日勤と休みを定めるだけの問題となっていた。しかし、フェーズ I と II では良い解が得られるものの、フェーズ III では、日勤と休みに対して自由度が残されておらず、うまいスケジュールを得ることができなかった。そこで、緩和問題を利用しないで、全ての種類のシフトを同時に考慮できる新しいアルゴリズムを作り上げる必要があった。

この章で構築されるアルゴリズムと、前章のアルゴリズムの各フェーズで利用したアルゴリズムとの違いは、部分問題軸アプローチの各イテレーションにおいて部分問題(5.7)(5.12)を解く際に、列挙のかわりに分枝限定法を利用するところにある。我々の計算実験で利用した分枝限定法の実装は、まだ基本しか考えられておらず、改善の余地を十分に残したものである。しかし、ここでの目的は、実行時間の速さではなく、良い解を得ることだと述べておきたい。これは、分枝限定法を、後でもっと効率のよいヒューリスティック・アルゴリズムと置き換えることで、速度向上できると考えるからである。

そこで、スケジューリングの期間を 1 週間の単位で切り分け(例えば、30 日なら、7 日 \times 4 と 2 日 \times 1 に分ける)それぞれの期間での各ナースの実行可能シフトパターンを全て列挙できるようにする。列挙される短いパターンは、その期間において、ナース拘束条件の Type II と Type III を満たすものとする。利用する分枝限定法の詳細について以下に述べる。(i)緩和問題：隣り合う期間のパターンに対し、ナース拘束条件の Type I と Type III を緩和したもの；(ii)分枝：次の期間のパターンの中で、現在の期間のパターンとの連結でナース拘束条件の Type III を満たすものに関して子問題を作成する；(iii)下界値：次の期間以降のパターンを全て評価して最も目的関数値を減らすパターンをそれぞれの期間で選ぶことにより緩和問題を解く；(iv)子問題選択：深さ優先方策を利用し、同じ深さにおいては最良パターンに対応する子問題を選択する；(v)上界値：アルゴリズムの各イテレーションにおいて、最初に解く部分問題に対しては上界値 $UB = \infty$ とし、その他の場合は、そのイテレーションですでに他の部分問題を解いて得られた目的関数値の最良の値を UB に設定する；(vi)テスト：実行可能解を持たない子問題を削除するために、ナース拘束条件の Type I をチェックする。

この分枝限定法を使ったアルゴリズムを以下に示すが、ここで分枝限定法は Step 8 において部分問題(5.7)(5.12)を解くことになる。

基本アルゴリズム(v0)

- Step 1. 各ナース $i \in M$ に対し、 $P_i = \phi$, $TABU_i = \phi$. $z^* = \infty$.
- Step 2. ダミー・パターン q_0 を作成。各ナース $i \in M$ に対し、 $q^i = q_0$.
- Step 3. $iteration = 1$.
- Step 4. $iteration > m$ ならば、各ナース $i \in M$ に対し、
 $TABU_i = \{q | removed_{iq} \geq iteration - TL, q \in P_i\} \cup \{q^i\}$.
- Step 5. 各ナース $i \in M$ に対し、 $iteration \leq m$ かつ $q^i \neq q_0$ ならば、 $lb_i = \infty$ 、そうでない場合、部分問題 i の緩和問題を解いて下界値 lb_i を得る。
- Step 6. lb_i の小さい順に部分問題を並べ、 i_1, i_2, \dots, i_m と表す。
- Step 7. $UB = \infty$. $\gamma = 1$.
- Step 8. $lb_{i_\gamma} < UB$ ならば、全ての $q \in TABU_{i_\gamma}$ を避けながら、上界値 UB 付き部分問題 i_γ を分枝限定法で解く。最適パターンを q_{i_γ} 、その目的関数値を z_{i_γ} とよぶ。 $z_{i_\gamma} < UB$ ならば、 $i^* = i_\gamma$ とし、 $UB = z_{i_\gamma}$.
- Step 9. $\gamma < m$ ならば、 $\gamma = \gamma + 1$ として Step 8 へ。
- Step 10. q_{i^*} を次の試行解に採用： $q' = q^{i^*}$, $q^{i^*} = q_{i^*}$, $z^* = z_{i^*}$. $q' \neq q_0$ ならば、 $P_{i^*} = P_{i^*} \cup \{q'\}$ とし、 $removed_{i^*q'} = iteration$.
- Step 11. $z^* = 0$ ならば終了。そうでない場合、 $iteration = iteration + 1$ とし、Step 4 へ。

この基本アルゴリズム(v0)に基づく以下の 4 つのパリエーションを考えた。1 つ目と 2 つ目のパリエーション(v1)と(v1.1)は、解かなくてはいけない部分問題の数を減らす工夫をする。Step 8 で現在の試行解の目的関数値より小さい値を持つ解が見つかったら、その解を次の試行

解として採用してしまい、そのイテレーションの残りの部分問題は解かないことにする (v1) は、この方策をダミー・パターンが取り除かれてからおこない (v1.1) は、最初のイテレーションからおこなう。3 目と 4 目のパリエーション(v2)と(v2.1)は、ダミー・パターンが取り除かれた後の Step 8 で別の分枝限定法を利用する以外は、それぞれ(v1)と(v1.1)と同じである。ここでの分枝限定法は、ストップング・ルールを追加したもので、現在の試行解の目的関数値より小さい目的関数値を持つ解が得られたら終了する。

最初の予備実験の結果からは、ダミー・パターンが取り除かれるまでは、各イテレーションで全ての部分問題を解く(v0)(v1)(v2)の方が、各イテレーションで 1 つしか部分問題を解かない(v1.1)(v2.1)よりも効率が良いことがわかった。しかし、これらのアルゴリズムの中では、(v2)と(v1)が最も効率よかったので、ダミー・パターンが取り除かれた後は、部分問題を解くアルゴリズムは最適解法である必要がないこともわかった。このことは、各イテレーションで部分問題をうまいヒューリスティックで解くようなアルゴリズムの構築の可能性を示している。従って、これらのアルゴリズムの速度向上が望めることになる。Ikegami-2shift-DATA1 に対して、DELL PowerEdge (Pentium III, 1 GHz, Linux 7.1) 上で、 $TL=10,000$ と設定した (v2) を使って解いたとき、実行時間が 135 分 36 秒 (120,460 イテレーション) と、前章のアルゴリズムに比べて非常に多くの時間を要していたものの、全ての拘束条件を満たす解を得ることができた。

次に、別の 2 交替データ (Millar らの Problem No. 1, $m=8, n=14$ についてサイクリックでないスケジュールを求める問題 (Millar and Kiragu, 1998)) を (v2) を使って解いた。ネットワーク・プログラミングによって得られた Millar らのスケジュールは、できれば避けたい 1 日だけの孤立勤務を 3 か所含んでいたが、我々は全ての拘束条件を満たしたスケジュールを得ることができた。実行時間は、DELL PowerEdge 上で、1 秒だけ (69 イテレーション) だったが、Millar らの IBM RISC6000/340 workstation 上での 502 秒とは、実行環境が異なるので直接には比較できないと考える。我々と Millar らの結果を表 8 と表 9 に示す (Millar らの問題の条件については Appendix A.5 を参照)。

最後に、3 交替の 3 つのデータを解いてみた。Ikegami-3shift-DATA1 (池上 他, 1996) は、主任ナースが提供してくれた実際のデータであり、DATA1.1 と DATA1.2 は、主任ナースとの議論に基づき、可能ならば満たしたいというナース拘束条件を、DATA1 にいくつか加えて計算実

表 8. 問題 No. 1 (Millar and Kiragu, 1998) に対する我々の結果 .

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	Off	TD	TN	Weekend off
1				TD	TD				TD	TD		TN	TN	TN	7	4	3	1
2	TN	TN			TD	TN	TN			TD	TD				7	3	4	1
3	TD	TD	TN			TD	TD				TN	TN			7	4	3	1
4	TD	TD	TD			TN	TN			TN	TN				7	3	4	1
5				TD	TN			TN	TN			TD	TD	TD	7	4	3	1
6				TD	TN	TN			TN	TN			TD	TD	7	3	4	1
7	TN	TN				TD	TD	TD			TD	TN			7	4	3	1
8				TN	TN			TD	TD			TD	TN	TN	7	3	4	1

表 9. 問題 No. 1 (Millar and Kiragu, 1998) に対する Millar らの結果 .

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	Off	TD	TN	Weekend off
1			TN	TN		TN	TN		TD	TN		TD			7	2	5	1
2	TN	TN			TD					TN	TN		TD	TD	7	3	4	1
3					TN	TN	TN		TD	TD	TD	TD			7	4	3	1
4			TN	TN		TD	TD	TN			TN	TN			7	2	5	1
5	TN	TN				TD	TD	TD			TD	TN			7	4	3	1
6			TD	TD	TN			TN	TN				TD	TD	7	4	3	1
7	TD	TD	TD					TD	TN				TN	TN	7	4	3	1
8	TD	TD		TD	TD				TD				TN	TN	7	5	2	1

さらに、DATA1 に対して最大実行時間を 100 時間に設定したところ (wcsp) の z^* は 13 のままだったが (v2) は $z^* = 6$ の解を実行開始後 5783 分 1 秒 (2,761,548 イテレーション) に得ることができた。結果のスケジュールを表 10 に示す。この解では、合計 90 シフトの中で、6 つのシフト (2 つの日勤, 1 つの準夜勤, 3 つの深夜勤) にのみ、ベテランナースが不足している。比較的重要でないナース拘束条件をわずかに緩和 (例えば、深夜勤は 2 連続が好ましいものの、日勤や準夜勤の直後に 1 回で終了する深夜勤を許し) ながら、紙の上でスケジュールを手直したとき、4 つの夜勤 (1 つの準夜勤, 3 つの深夜勤) に関するシフト拘束条件の全てについて、考慮することができた。

我々のスケジュールは、主任ナースの作成した実際のスケジュール (24 のシフトにおける合計 39 のシフト拘束条件と 71 のナース拘束条件を満たしていないもの) に比べて、はるかに好ましいものとなっていることがわかった。

もし可能であるならば、Ikegami-3shift-DATA1, DATA1.1, DATA1.2 を完全に解くことは、組合せ最適の観点からも非常に興味深い問題であると考えられる。

8. おわりに

この研究では、ナース勤務表作成の現場調査を徹底的におこない、実際に考慮されている勤務表作成の条件を整理することにより、日本のナース・スケジューリングを扱うことができる数理計画モデルを構築した。そして、部分問題軸アプローチを提案し、このアプローチに基づくアルゴリズムが良い解を与えることを示した。予備計算実験の結果は 2 交替と 3 交替を区別することなく問題を扱える効率のよいアルゴリズムの存在の可能性を示した。本論文では、これら「問題解決の全過程」を具体的に示すことができたと考えられる。

7 章で提案したアルゴリズムで作成した勤務表は、東京女子医科大学附属病院の 2 交替制部署 (チーム数 3, ナース数 35) の 2003 年 1 月の勤務表のたたき台になった。しかし、まだ我々のアルゴリズムは「誰にでも容易に利用できるインタフェース」を持っておらず、現場の勤務表作成担当者が直接利用することができない。現場で実際に考慮されている条件を、誰にでも容易に入力でき、アルゴリズムを実行できるようなインタフェースを持つ支援システムの構築が急務であると考えられる。これと平行して今後も、さらに有効なアルゴリズムを構築するために、部分問題を効率よく解けるヒューリスティック・アルゴリズムの開発に力を注ぎたい。

部分問題軸アプローチとそのアルゴリズムは、日本や海外のナース・スケジューリングだけでなく、医療、介護はもちろん、ビジネスやロジスティクスのエリアの他の多くの問題にも適用できると考える。特に、医師の当直のスケジューリングや、レストラン、スーパーマーケット、テーマパーク等のアルバイトのスケジューリングのように、スキルレベルを考慮しなければならぬスタッフ・スケジューリングに対し、このアプローチは有効であると考えられる。現在、訪問介護事業所等の増加に伴い、常勤よりもパート型や登録型といった形での勤務が増えている。このように非常勤者を含むスタッフ・スケジューリングにおいては、スタッフの勤務可能時間帯が限られている場合が多く、解の探索空間が縮小され、ナース拘束条件にあたる条件も大幅に少なくできることから、ナース・スケジューリングに比較して、非常に速いスピードで解を得ることが可能である。同じく、ナース拘束条件にあたる条件が比較的少ない医師の当直スケジューリング (Ikegami, 2004) では、7 章のアルゴリズムを利用して 1 ヶ月分の当直表を非常に短い時間 (1 秒以下 ~ 数 10 秒) で解くことに成功している。

一方、ナース・スケジューリングの他にも、列車の乗務員計画等、複雑な条件を含むスタッフ・スケジューリングが現実の世界には存在する。しかし、多くの場合、勤務表作成が不可能な場合には予備要員を加えて作成することになる。予備要員の利用を想定しない現在の日本の

ナース・スケジューリングは、スタッフ・スケジューリングの中でも最も難しい部類の問題であり、この問題の解決は世の中の非常に多くのスタッフ・スケジューリングならびに同じ構造を抱える問題の解決に貢献するものと考える。

謝 辞

本論文について貴重なご助言を下された査読委員ならびに編集委員の先生方に深く感謝致します。この研究の一部は、文部科学省科学研究費補助金(C)No. 12672295 (C)No. 16510120)、文部科学省ハイテクリサーチセンター整備事業研究費、統計数理研究所共同利用研究費(16-共研-2016)を受けておこなわれました。

Appendix : データ (<http://cleo.is.seikei.ac.jp/~atsuko/>でも参照可能)

A.1 Ikegami-2shift-DATA1(2章の表1の勤務表作成データ)

シフト拘束条件

シフト拘束条件について述べた後、それらを「各グループの所属ナースが働く数」の下限値と上限値で表現し、表11に示す。

表 11. グループ毎の働くナース数の下限値・上限値(2shift-DATA1)。

(- : 日勤, Nn : 夜勤, L : 下限値, U : 上限値)

Group	All	A		B		C		Skilled		A-ss		B-ss		C-ss		RQ			
Member (No.)	all 1,...,28	1,...,10		11,...,19		20,...,28		1,2,3,11,12,20,21		1,...,7		11,...,17		20,...,26		8,9,10,18,19,27,28			
Shift	-	Nn		-	Nn		-	Nn		-	Nn		-	Nn		-	Nn		
Bound	L	U	L	U	L	U	L	U	L	U	L	U	L	U	L	U	L	U	
1	F	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
2	S	9	10	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
3	S	9	9	4	4	3	3	1	2	3	3	1	2	3	3	1	2	3	3
4	H	9	9	4	4	3	3	1	2	3	3	1	2	3	3	1	2	3	3
5	T	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
6	W	12	14	4	4	4	5	1	2	4	5	1	2	4	5	1	2	4	5
7	T	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
8	F	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
9	S	9	10	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
10	S	9	9	4	4	3	3	1	2	3	3	1	2	3	3	1	2	3	3
11	M	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
12	T	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
13	W	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
14	T	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
15	F	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
16	S	9	10	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
17	S	9	9	4	4	3	3	1	2	3	3	1	2	3	3	1	2	3	3
18	M	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
19	T	12	14	4	4	4	5	1	2	4	5	1	2	4	5	1	2	4	5
20	W	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
21	T	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
22	F	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
23	H	9	10	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
24	S	9	9	4	4	3	3	1	2	3	3	1	2	3	3	1	2	3	3
25	M	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
26	T	12	16	4	4	4	6	1	2	4	6	1	2	4	6	1	2	4	6
27	W	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
28	T	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
29	F	10	11	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4
30	S	9	10	4	4	3	4	1	2	3	4	1	2	3	4	1	2	3	4

夜勤のシフト拘束条件

- 毎日の夜勤に働くナースの合計数は 4 名必要．
- A, B, C 各チームからベテラン(skilled)もしくは 2 年目(second-year)ナースが少なくとも 1 名, 全体でベテランが少なくとも 1 名必要．

日勤のシフト拘束条件

- 毎日の日勤について, 各チームからベテランもしくは 2 年目ナースが少なくとも 2 名, 全体でベテランが少なくとも 1 名必要．
- 通常のウィークデイは, 合計で 10~11 名, 各チーム 3~4 名必要．
- 日曜日と祝祭日(4 日と 23 日)には合計 9 名, 各チーム 3 名ずつ必要．
- 6 日と 19 日は, 合計 12~14 名, 各チーム 4~5 名必要．
- 26 日は, 合計 12~16 名, 各チーム 4~6 名必要．

ナース拘束条件

表 12 は各ナースの休み, 日勤, 夜勤の回数の下限値と上限値(Type I 拘束条件)を示す．

表 13 は前月末からのスケジュールとナース達の勤務や休みの希望(Type II)を示す．この他の制約(Type III, 他)は以下の通り：

- 各ナース, 少なくとも 1 回は週末 2 連休(2~3 日, 3~4 日, 9~10 日, 16~17 日, 23~24 日)の中から選ぶ．4 日は祭日)を必要とする．
- 夜勤の翌日は休みでなければならない．
- 夜勤と夜勤の間は少なくとも 3 日あけなければならない．
- 1 週間に 1 回は休みが入らなくてはならない．
- 4 日連続の日勤は許されていない．
- 日勤・休み・日勤・休み・日勤というパターンは許されていない．
- 前月末とのシフトの並びについても考慮しなければならない．

A.2 Ikegami-3shift-DATA1

表 14 はシフト拘束条件, 表 15 ~ 18 はナース拘束条件を示す．

A.3 Ikegami-3shift-DATA1.1

Ikegami-3shift-DATA1 に加え, 以下のナースと日にち(ナース i , 日 j)について夜勤を避ける (2,8) (2,17) (3,1) (4,10) (4,24) (5,21) (6,15) (8,24) (9,11) (9,22) (11,8) (12,6) , (13,8) (13,23) (14,26) (15,17) (16,19) (17,19) (18,9) (19,22) (20,8) (20,17) (21,7) , (22,8) (23,8) (24,1) (24,24) .

A.4 Ikegami-3shift-DATA1.2

Ikegami-3shift-DATA1.1 に加え, ナース 19 について, 1~3, 6, 8~10, 13~15, 17, 20~22, 27~30 日にあたる準夜勤と深夜勤を避ける．

A.5 問題 No.1(Millar and Kiragu, 1998)

この病院では, 12 時間シフトで稼働している(TD: 日勤, TN: 夜勤) . ナースは合計 8 名で, 各シフト 2 名ずつ必要とされている . 各ナースは 2 週間で高々 7 シフト勤務する . 各ナースは, 2 回の週末のうち, 少なくとも 1 回は連休を必要とする . 連続勤務は 4 日までとし, シフトとシフトの間は少なくとも 12 時間あけ, 夜勤は 3 連続までしかできない . 可能ならば, 1 日だけの孤立勤務を避ける . 4 日連続勤務も避けることが望まれる . もし, 4 日連続勤務が入ってしまった場合には, 少なくとも 2 日間の休みが必要である .

表 14. グループ毎の働くナース数の下限値・上限値(3shift-DATA1).

(- : 日勤, e : 準夜勤, n : 深夜勤, L : 下限値, U : 上限値)

Group	All	A	B	A-ss	B-ss	B-ss+s'	B-rq+s'
Member (No.)	all	1,...,13	14,...,25	1,...,6	14,...,18	14,...,18,25	19,...,25
Shift	- e n	- e n	- e n	- e n	- e n	- e n	- e n
Bound	L U L U L U L U L U L U	L U L U L U L U L U L U L U	L U L U L U L U L U L U L U	L U L U L U L U L U L U L U	L U L U L U L U L U L U L U	L U L U L U L U L U L U L U	L U L U L U L U L U L U L U
1	W	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 3 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
2	T	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
3	F	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
4	S	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
5	S	7 11 4 4 3 3 3 6 2 2	1 2 3 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 2 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
6	M	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
7	T	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 3 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
8	W	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
9	T	7 11 4 4 3 3 3 6 2 2	1 2 3 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 1 2 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
10	F	7 11 4 4 3 3 4 6 2 2	1 2 3 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 1 2 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
11	S	7 11 4 4 3 3 4 6 2 2	1 2 3 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 1 2 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
12	S	7 11 4 4 3 3 3 6 2 2	1 2 3 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 2 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
13	M	7 11 4 4 3 3 4 6 2 2	1 2 3 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 1 2 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
14	T	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 3 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
15	W	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
16	T	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 3 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
17	F	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 3 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
18	S	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
19	S	7 11 4 4 3 3 3 6 2 2	1 2 3 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 2 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
20	M	8 11 4 4 3 3 4 6 2 2	1 2 3 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 1 2 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
21	T	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 5 1 1 0	2 0 2 0 1 0 1	
22	W	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
23	T	13 18 4 4 3 3 6 9 2 2	1 2 5 9 2 2 1 2 2 2	6 1 1 1 1 6 1 1 0	1 2 6 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
24	F	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 3 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
25	S	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 3 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
26	S	7 11 4 4 3 3 3 6 2 2	1 2 3 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 2 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	
27	M	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
28	T	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
29	W	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	3 1 1 1 1 3 1 1 0	1 2 3 1 2 1 1 2 6 1 1 0	2 0 2 0 1 0 1	
30	T	8 11 4 4 3 3 4 6 2 2	1 2 4 6 2 2 1 2 2 2	1 1 1 1 1 2 1 1 0	1 2 3 1 2 1 1 2 4 1 1 0	2 0 2 0 1 0 1	

表 15. シフトの連続日数と間隔の下限値・上限値(3shift-DATA1).

	連続日数		間隔日数	
	LB	UB	LB	UB
/ 休み	1	5	2	6
d 日勤	1	4	1	6
e 準夜勤	1	3	1	∞
n 深夜勤	2	2	6	∞

表 16. 許されないシフトの並び(3shift-DATA1).

n -	n +	n e	e -	e +	n / -	n / +
-----	-----	-----	-----	-----	-------	-------

参 考 文 献

- Aickelin, U. and Dowsland, K. A. (2000) Exploiting problem structure in a genetic algorithm approach to a nurse rostering problem, *Journal of Scheduling*, **3**, 139–153.
- Arther, J. L. and Ravindran, A. (1981) A multiple objective nurse scheduling model, *AIIE Transactions*, **13**, 55–60.
- Bell, P. C., Hay, G. and Liang, Y. (1986) A visual interactive decision support system for workforce (nurse) scheduling, *INFOR*, **24**, 134–145.
- Chan, L. K., Falkenberg, J. and Rosenbloom, E. S. (1987) Implementation problems of a nurse preference mathematical programming approach to scheduling, *Congressus Numerantium*, **56**, 251–260.
- Chvatal, V. (1983) *Linear Programming*, W. H. Freeman and Company, New York.
- Dowsland, K. A. (1998) Nurse scheduling with tabu search and strategic oscillation, *European Journal of Operational Research*, **106**, 393–407.
- Dowsland, K. A. and Thompson, J. M. (2000) Solving a nurse scheduling with tabu search and strategic oscillation, *Journal of the Operational Research Society*, **51**, 825–833.
- 池上敦子 (2000) 2 交替制ナース・スケジューリングのアルゴリズム改善, *Journal of the Operations Research Society of Japan*, **43**, 365–381.
- 池上敦子 (2001) 我が国におけるナース・スケジューリング: モデル化とアプローチ, 博士論文, 成蹊大学工学部, 東京.
- 池上敦子 (2004) ナース・スケジューリング, RAMP シンポジウム論文集, 1–16.
- Ikegami, A. (2004) Scheduling doctors for night duty, *Proceedings of the 29th Meeting of the European Working Group on Operational Research Applied to Health Services*, 135–145, Faculty of Informatics and Statistics, University of Economics, Prague & Czech Society for Operational Research.
- 池上敦子, 丹羽 明 (1998) ナース・スケジューリングに有効なアプローチ: 2 交替制アルゴリズムにおける実現, *Journal of the Operations Research Society of Japan*, **41**, 572–586.
- Ikegami, A. and Niwa, A. (2003) A subproblem-centric model and approach to the nurse scheduling problem, *Mathematical Programming*, **97**, 517–541.
- 池上敦子, 相澤 学, 大倉元宏, 若狭紅子, 松平信子, 越河六郎 (1995) ナース・スケジューリング・システム構築のための基礎的調査研究, *労働科学*, **71**, 413–423.
- 池上敦子, 丹羽 明, 大倉元宏 (1996) 我が国におけるナース・スケジューリング問題, *オペレーションズ・リサーチ*, **41**, 436–442.
- Jaumard, B., Semet, F. and Vovor, T. (1998) A generalized linear programming model for nurse scheduling, *European Journal of Operational Research*, **107**, 1–18.
- Kostreva, M. M. and Jennings, K. S. B. (1991) Nurse scheduling on microcomputer, *Computer & Operations Research*, **18**, 731–739.
- Li, L. X. and Benton, W. C. (1996) Performance measurement criteria in health care organization: Review and future research directions, *European Journal of Operational Research*, **93**, 449–468.
- Megeath, J. D. (1978) Successful hospital personnel scheduling, *Interfaces*, **8**, 55–59.
- Meyer auf'm Hofe, H. (2001) Nurse rostering as constraint satisfaction with fuzzy constraints and inferred control strategies, *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, **57**, 67–99.
- Millar, H. H. and Kiragu, M. (1998) Cyclic and non-cyclic scheduling of 12 h shift nurses by network programming, *European Journal of Operational Research*, **104**, 582–592.
- Miller, H. E., Pierskalla, W. P. and Rath, G. J. (1976) Nurse scheduling using mathematical programming, *Operations Research*, **24**, 857–870.

- Musa, A. A. and Saxena, U. (1984). Scheduling nurses using goal-programming techniques, *IIE Transactions*, **16**, 216–221.
- 長野弘志, 宮崎茂次 (1996). 勤務に対する嗜好の個人差を考慮した看護婦スケジューリング: 遺伝アルゴリズムによる看護婦スケジューリング(第一報), *日本経営工学会論文誌*, **47**, 143–149.
- Nonobe, K. and Ibaraki, T. (1998). A tabu search approach to the constraint satisfaction problem as a general problem solver, *European Journal of Operational Research*, **106**, 599–623.
- Osogami, T. and Imai, H. (2000). Classification of various neighborhood operations for the nurse scheduling problem, *Lecture Notes in Computer Science*, **1969**, 72–83.
- Ozkarahan, I. and Bailey, J. E. (1988). Goal programming model subsystem of a flexible nurse scheduling support system, *IIE Transactions*, **20**, 306–316.
- Randhawa, S. U. and Sitompul, D. (1993). A heuristic-based computerized nurse scheduling system, *Computer & Operations Research*, **20**, 837–844.
- Rosenbloom, E. S. and Goertzen, N. F. (1987). Cyclic nurse scheduling, *European Journal of Operational Research*, **31**, 19–23.
- 嶋田葉子, 池上敦子, 大倉元宏 (2001). 看護婦勤務表作成支援システムの開発を意図したタスク分析, *人間工学*, **37**, 125–133.
- Siferd, S. P. and Benton, W. C. (1992). Workforce staffing and scheduling: Hospital nursing specific models, *European Journal of Operational Research*, **60**, 223–246.
- Smith-Daniels, V. L., Schweikhert, S. B. and Smith-Daniels, D. E. (1988). Capacity management in health care services. Review and future research directions, *Decision Sciences*, **19**, 889–919.
- Warner, D. M. (1976). Scheduling nursing personnel according to nursing preference: A mathematical programming approach, *Operations Research*, **24**, 842–856.

Nurse Scheduling
—Site Research, Modeling and Algorithms—

Atsuko Ikegami

Faculty of Science and Technology, Seikei University

This paper deals with the scheduling of nurses to staff shifts at a hospital. The basic difficulty lies in the necessity of maintaining a certain level of service and skill in the makeup of every shift, while balancing the workload among the nurses involved. As a result, it is usually impossible to develop a schedule that satisfies all the requirements in spite of the time and resources spent in the effort.

In this paper, we present our latest developments by demonstrating the entire process of problem solving, from modeling based on site research and a survey of possible applications. We first describe the nurse scheduling problem in Japan, and then present our model that accounts for the problems identified in Japanese scheduling systems. We propose an efficient approach that takes advantage of the structure clarified in the modeling, and based on this approach, we show an algorithm for a 2-shift system that provides efficient, appropriate solutions. Finally, to assess the possibility of using efficient algorithms that can account for both 2-shift and 3-shift systems, we show the results of preliminary computational experiments using newly developed, prototype algorithms. The proposed model and approach can be adapted for the majority of hospitals in Japan, as well as for some hospitals in other countries and could be applied to many other staff scheduling problems in the fields of health services, business and logistics.