

運用コストを重視した最適化

—小規模な事業所で運用可能なシステムを考える—

池上 敦子, 宇野 毅明, 足立 幸子, 村野 真悟, 佐藤 広幸,
吉田 勇人, 軍司 奈緒, 内山 広紀

1. はじめに

近年, OR, 特に最適化の応用は, IT 技術の発達に伴い, その範囲を広げていると考えられる。しかし, その一方で OR 論文における事例研究の数は, それに伴うほどは増えておらず, 最適化をはじめとする OR の技術が前面に出たシステムの開発もまだ多くはない。その理由を考えて, OR の技術が低いというわけではなく, 例えば近年の機械学習分野では, 比較的古くに開発された連続最適化や劣モジュラ最適化の技術が最新の研究で使われており, OR の技術の高さを見ることができ。考えるに, 実分野それも比較的小さな事業単位への応用の少なさは, その運用コスト, つまり計画立案を行い, それを実行するためのコストの高さにありそうである。いくら優れた技術であっても, その導入や日々の利用に多大なコストがかかるようでは, 利用するモチベーションを高めることはできない。実際, 過去の研究でもナーススケジューリング問題 [1] をはじめとする比較的現場に近い問題に取り組んできたが, 運用コストの高さのために実用化に至っていないものが多く, 反省すべきところが多々あると考えている。

以降, 運用コストを減少させるという観点で議論を進めるが, ひとえに運用コストと言っても, その種類は多岐にわたり, また場面ごとにその性質は異なる。例えば, 製鉄所での板取り問題や小売店への配送計画などでは基幹データベースと作成するシステムをどのように連携させるか, という点が大きな課題となる。場

合によっては, 基幹データベースの一部として作り込む必要もある。また, 現状を自動的に把握するためのセンサー類, 通信装置などのインフラが必要となることなど, システム以外の投資が必要となる場合もある。また, プログラムの実装, あるいは既存システムのカスタマイズも大きな負担となる。大企業の場合, 問題が多様な条件を持つことが多く, それを明文化するのは非常に骨の折れる仕事である。このような場面では, 運用コストはつまりお金の問題であり, 投資に見合う分のコストダウンが見込まれれば利用する, ということになる。よって, システム開発を得意とするソフトウェアメーカーが, カスタマイズの比較的容易な経理・事務系の作業を中心に IT 化をするといったことが多くなる。このような状況では, 個性あふれる多種の問題に対して先端技術で解決法を示すというアプローチは, 場合によって金銭的なデメリットを抱えることにもなる。

では, 上記とは対象に, 運用コストがお金だけではない分野はどうであろうか。例えば, 訪問介護ヘルパーの勤務表(勤務スケジュール)を最適化する訪問介護スタッフスケジューリングの場合, 運用コストはお金の問題だけではない。スタッフであるヘルパーそれぞれの希望や可能性を収集し, それをシステムに入力する手間, 作成した勤務表を実態に合わせるため, システムをカスタマイズする, あるいは勤務表を実態に合わせて修正する手間, 変更があったときに勤務表を作り直す手間, できた勤務表をヘルパーと利用者の両方に周知する手間などがあり, むしろ金銭的な問題よりも人件費として換算しにくい手間の問題のほうが大きい。基幹データベースが存在するなら, それに合わせたデバイスや実装を行う必要があり, そのコストを下げることは容易ではない。しかし, 人間はある程度融通が利くため, 作業内容を変えることによるコストの減少は可能であろう。例えば Web 検索では, ユーザは

いけがみ あつこ, あだち さちこ, むらの しんご,
さとう ひろゆき, よしだ はやと, ぐんじ なお,

うちやま ひろのり

成蹊大学理工学部

〒180-8633 武蔵野市吉祥寺北町 3-3-1

うの たけあき

国立情報学研究所

〒101-8430 千代田区一ツ橋 2-1-2

受付 12.5.14 採択 12.9.21

目的のページを見つけるために入力キーワードを工夫する。これは機械に目的を理解させるという運用コストが高い作業をあきらめ（現在の技術では多くのデータ入力が必要であろう）人間にその作業を委託し、代わりに迅速な計算や検索結果の上手な可視化に重点を置いている。このように、システム化やモデルの作成に工夫を施すことで運用コストを下げることは、研究するに値することだと考えられる。

少し例を挙げよう。既存の最適化システムでは、解を修正する機能は充実していないことが多かった。しかし、運用コストという観点からは、修正機能を充実させることで、実態に合わせた変更や急な人員変化による変更のコストを下げるができる。この際、解をどのように可視化するか、という点が変更作業の効率に大きくかわってくる。モデルが人と人の関係をデータとして要求すると、人はそれを入力しなければならず、多大なコストがかかる。このようなデータは、なるべく使わない、簡単に入力できる（デフォルト値や一括入力）、過去のデータの再利用、一部のデータから残りを推測、といった機能があると、運用コストは大きく減少する。データ入力やカスタマイズの簡素化を考えるなら、場合によってはモデルの精度は犠牲にしなければならないかもしれない。モデルの精度と運用コスト、2つの目的を同時に達成するバランスのとれた解を求めることが、運用コストが低いシステムを作るうえで重要なことであろう。

本稿では、このような運用コストを減少するOR、特に最適化の問題について議論したい。ただし、一口に運用コストと呼ぶと、投資できる金額に論点が移ってしまうことが考えられるため、ここでは医療や介護現場、そしてサービス業といった比較的小規模な最適化の場面に限定し、そのような問題を小規模ビジネスORと呼ぶことにする。また、小規模ビジネスORの1つとして、われわれが開発した訪問介護スタッフスケジューリングのWebサービスシステムを紹介したい。このシステムを小規模ビジネスOR的な観点から評価し、運用コストを下げるという方針がどの程度現実的であるのか考えるとともに、今後のOR研究で小規模ビジネスORを研究する重要性について議論したい。

小規模な現場には、1つの最適化で得られる利得が小さいというデメリットもあるが、逆にいくつか興味深い側面もある。1つは、似た問題を抱える現場が数多くあるということである。例えば病院であれば、日本だけでも数千、問題を保有する単位である部署別に考えれば数万の現場がある。ただし、これらの現場はす

べて均質なわけではなく、それぞれ固有の要件を持っている。このような状況で、運用コストの小さいシステムを構築するには、いかに各病院の共通する要件を抽出してモデル化を行うか、いかにカスタマイズのコストが小さいモデルを作るか、という点が重要になる。また、小規模な事業では、柔軟な経営や運用を行うことで業務の質を上げていることが多く、計画立案を行う人間の創意工夫が果たす役割が大きい。そのため、完全な自動化を行って人間の意志の介在を排除してしまうと、柔軟性の損失というある種の運用コストが発生してしまう。このことは、われわれの訪問介護事業所に対する調査から観察された「既存ITシステムの最適化機能（スケジューリング機能）の利用率低さ」からも裏付けられている[4, 5]。創意工夫は暗黙知に基づいて行われており、それをすべてコンピュータ上でモデル化することは不可能に近い。そのため、解の上手な可視化と操作性の良い修正手段が提供されることが必要となる。

2. 小規模ビジネスORの方向性

本節では、小規模ビジネスORで考慮する運用コストにはどのようなものが考えられるか議論する。ただし、これは小規模ビジネスORを定義するものではない。小規模ビジネスORは、OR研究において多少置き去りにされた問題¹であり、新しいアルゴリズムの提案対象となりにくく、それ以外の扱いが困難な問題を表すものとして話を進めるが、将来的にその要件は変わって行くであろう。ここでは現時点でわれわれが認識している点を挙げることにとどまりたい。

まず、小規模ビジネスの想定であるが、病院の一部署やレストランなど、スタッフが10人から100人程度の規模を想定している。しかし、これは絶対的なものではない。問題の規模（変数・制約の数）は膨大にはならず、具体的には、操作をするユーザに対して可視化が可能であり、全体を俯瞰しながら人間が解を修正・構築できる程度の規模を想定する。この要件は、「人間にある程度の部分を任せる」という方針からは、必然的なものとなる。

小規模ビジネスORでは、最適化問題を中心として考えることが多いと思われる。理由の1つは、AHPやDEAといった分野では、その中心的な応用はすでにある程度小規模で、人間が把握可能な可視化が可能であり、すでに小規模なビジネスを想定せずとも、す

¹ もちろん、近年、勤務スケジュール作成の研究が盛んである[2, 3]が、本稿で提案する視点での研究はまだない。

に小規模ビジネス OR の要件を満たす研究が多く存在するからである。逆に言えば、運用コスト最小化という概念は、最適化に適しているという側面もある。そのため、以下では主に最適化を中心として、運用コストやその評価方法を議論したい。

ここで考える運用コストは、システムを利用するための金銭的なコスト（開発費、利用費など）、労力（導入、日々運用）、以前の解からの変化による現場での損失（既存の作業からの変化）、システム導入による従業員の習熟のためのコストや心情の変化、ストレスなどを含む。また、解を実利用するために必要とされる要件（暗黙知であることが多い）を満たさない場合にかかるデメリット、あるいはペナルティのようなものも運用コストとして考慮している。具体的な項目を以下に挙げる。

導入：システムの開発費、あるいは価格。導入にかかる手間や専門知識の必要性など。既存システムとの親和性も含む。また、コンピュータを含む新規のデバイスを購入するコストもここに入る。一回導入したシステムを繰り返し使うことで導入コストが相対的に下がる点も考慮する。さらに、ユーザのソフトウェアの習熟にかかるコストも考慮する。

データ入力：作業者がシステムにデータ入力をする労力。デバイスや消耗品を使う場合は、そのコストも考慮する。

連絡：解に基づいて作業をする勤務者との連絡コスト。例えば、現場の状況の連絡、休みの希望などを勤務者・顧客から連絡し、作成した解を元に勤務者や顧客への指示を送るためのコスト。

勤務の質：特定の勤務者・部署に負担が集中していないか、無意味に見える作業を強いられないか、急な変更能耐えられるか、などを考慮する。

計画：計画立案・分析・最適化にかかるコスト。安価なシステムで、実用的な時間内に求解できるか。また、求めた解を現場に即するように変更するために、どの程度労力が必要か [6]、など。

透明性：モデルに納得感があるかの評価。なぜその解が求めたか、納得がいく結果（解作成の方針が理解できる）か、そして、パラメータをはじめとするシステムの運用の引継が容易か、など。

創意工夫：求めた解に自身の考えで変更を加えることは可能か。また制約条件などの状況把握が容易で、自身の操作が与える結果が直観的にわかりやすいか。

本稿では、これらの観点から運用コストを評価することを提案したい。ただし、定量化が困難であるもの

が多く、モデル化を行うときにどのように考慮したかという点と、実際に利用したユーザの作業時間の変化や感想を聞く、といった評価方法が主体となる。

3. 訪問介護とスケジュールリング

訪問介護は、在宅で介護サービスを必要とする人にサービスを提供するためにスタッフを派遣する業種である [7]。近年介護保険制度の導入によりその規模が拡大したが、一方で慢性的な労働力不足を招いている。また、国の定めた基準どおりに物事を進める必要が発生し、業務の柔軟性が失われた点もある。利用者サービスを提供する時刻は月によって大きく変化し、偏りがある一方で、ヘルパーは登録型の勤務形態を取ることが多く、時間的な条件が厳しい。このような状況で、介護の質を守りつつ、かつサービスを円滑に提供するためには、ヘルパーの勤務スケジュールを上手に作る必要がある不可欠である。

過去（2004年）に行った調査 [4] では、勤務表は多くの事業所で手作業で作られていた。なんらかの IT システムを持つところもあったが、会計事務中心のシステムのために、前節で述べた運用コストがとても高く、スケジュールリング機能を実際に使っているところは非常に少なかった。一方で、ヘルパーの数、利用者の数ともに 100 人を超えるような事業所も存在し、勤務表の作成は人手で行うには大きすぎるコストが内在している。われわれの調査では、勤務表作成に月にのべ 100 時間以上を費やしている事業所も存在した。本稿では、なるべく多くの事業所でそのまま利用できるようなシステムを目指し、問題のモデル化とアルゴリズムの構築を行い、上記運用コストが下がるようなインタフェースをデザインする。

モデル化は、2003 年より毎月 1~2 回のペースで行ってきた訪問介護事業所におけるスケジュール作成の観察とインタビュー、そしてアンケート調査の結果を基に行い、どの事業所でも必ず考慮している実行可能性に関わる以下の内容を制約条件とした。

- ・ヘルパーの勤務に関わる制約
 - ヘルパーは自身の勤務可能時間帯にサービスを提供する
 - ヘルパーは同時刻に 2 つ以上のサービスを提供しない
- ・サービスに関わる制約
 - 利用者は希望するサービスをちょうど 1 人のヘルパーから受ける
 - サービスを提供するヘルパーは、サービス提供に

必要な資格・能力を持つ

・担当者に関わる制約

- 各利用者のサービスを行うヘルパーは、あらかじめ決められた担当ヘルパーの中から選ばれる
- 特定のサービスに対し、あらかじめヘルパーが指定されている場合、そのヘルパーが優先的に割当てられる

・ヘルパーの移動時間に関わる制約

- ヘルパーの提供するサービスの間には、サービスを提供する利用者の家を移動するための十分な空き時間が確保される

また、調査では多くの事業所で、勤務時間量が公平に、つまり給料が公平になるように勤務表を作成していた。しかし、各ヘルパーの勤務可能時間量が不均一であるため完全な公平は難しく、ほぼ実現不可能であることに加え、個々に適正な勤務時間量があると考えられているため、勤務時間量に上下限を設定することで対応した。また、できれば守りたい上下限を設定し、その制約を違反する度合いを目的関数で最小化することにして、公平性を担保することとした。

既存研究 [8, 9] でのヘルパーごとの実行可能スケジュールを組合せた定式化では、部分スケジュールをあらかじめ列挙することなどの計算コストが高くなる危険がある。そこで、割当問題もしくは輸送問題の構造を意識し、ネットワーク用の線形計画法を利用しやすい以下の定式化を利用することにした。

ヘルパーの集合を H 、サービスの集合を S とする。意思決定変数としては、ヘルパー i がサービス s を扱うか否かを表す x_{is} を利用する。そして、サービス s の開始時刻を b_s 、終了時刻を e_s 、担当可能なヘルパーの集合を H_s とし、サービス s の利用者宅とサービス t の利用者宅の間の移動時間を d_{st} とする。ヘルパー i の勤務時間量についてできれば守りたい下限と上限をそれぞれ l_i と u_i 、厳密な上下限を l'_i と u'_i とする。そして、下限 l_i に満たない時間量を α_i 、上限 u_i を超す時間量を β_i で表し、それらに対するペナルティを、それぞれ w_i^- 、 $w_i^+ \geq 0$ とする。さらに、すべてのサービスをカバーできない現実に対応させるため、カバーできないときに 1 となる変数 γ_s を用意し、目的関数で、 $(e_s - b_s)$ に比例する絶対的に大きな重み W_s をかけて最小化する。また移動時間制約を表現するために十分大きな数 M を利用する。

定式化

$$\text{Minimize } \sum_{s \in S} W_s \gamma_s + \sum_{i \in H} w_i^- \alpha_i + \sum_{i \in H} w_i^+ \beta_i \quad (1)$$

subject to

$$\sum_{i \in H_s} x_{is} + \gamma_s = 1, \quad s \in S, \quad (2)$$

$$\sum_{s \in S} (e_s - b_s) x_{is} + \alpha_i \geq l_i, \quad i \in H, \quad (3)$$

$$\sum_{s \in S} (e_s - b_s) x_{is} - \beta_i \leq u_i, \quad i \in H, \quad (4)$$

$$l'_i \leq \sum_{s \in S} (e_s - b_s) x_{is} \leq u'_i, \quad i \in H, \quad (5)$$

$$(e_s + d_{st}) x_{is} \leq b_t + M(1 - x_{it}), \quad i \in H, s, t \in S, b_s \leq b_t, \quad (6)$$

$$x_{is} = 0 \text{ or } 1, \quad i \in H, s \in S, \quad (7)$$

$$\alpha_i, \beta_i \geq 0, \quad i \in H, \quad (8)$$

$$\gamma_s \geq 0, \quad s \in S. \quad (9)$$

目的関数 (1) では、カバーできないサービス時間量を最小化することを最優先し、その下で勤務量の上下限を違反する度合いを最小化する。制約式 (2) は、すべてのサービスがカバーされるためのもので、どうしてもカバーできない場合は $\gamma_s = 1$ となり、目的関数が大きく増加する。制約式 (3)(4)(5) は、勤務量の上下限、緩和上下限を考慮するものであり、制約式 (6) は、サービス時間の重なりや移動時間制約を考慮する。

4. 勤務スケジュール作成

前節の定式化は整数計画問題となり、解を得るには最適化汎用ソルバーの直接的利用も考えられるが、訪問介護事業所にはその価格、もしくはフリーソフトであればその理解とインストールが大きな負担となる。現場で簡単に利用できるシステムを作るためには、ある程度の作り込みも必要とされることから、自身でアルゴリズムをデザイン・実装することとした。そこで、ネットワーク流構造を十分に利用し、以下のように (6)(7) 式を緩和した問題を子問題として解くこととした。

各ヘルパー $i \in H$ と各サービス $s \in S$ に対してノードを作成し、サービス時刻制約、担当制約、ヘルパー勤務時間帯制約を満たすノード i と s の間にアーク (x_{is} に対応) を設定してアークコストを 0 とする。アークの流量は、サービスノードへの勤務時間量を表すこととし、流量の下限を 0、上限をサービスノードの時間量 $(e_s - b_s)$ とする。本来、流量は 0 もしくは $(e_s - b_s)$ 、

つまり、その流量を流すか否かのどちらかであり、その間の値は許されないが、ここでは(7)式の緩和として考えることにする。また、サービスをカバーできない場合に備え、ダミーヘルパーノードを作成し、各サービスノードにアーク(γ_s に対応)を設定して、アークコストを W とする。また、ダミーヘルパーノードから流量を受け取れるダミーサービスノードを作成し、アークコストが0、流量下限が0、上限がサービス時間の総和 $\sum_{s \in S} (e_s - b_s)$ となるアークを設定する。これらの結果として、ヘルパーノードの供給量を「働ける時間量」、サービスノードの需要量を「サービス時間量」を設定すれば、古典的な輸送問題となる。

次に、勤務時間量を考慮するために、スーパーノードを1つ作成し、そこから各ヘルパーノードにそれぞれ4本のアークを設定して、勤務時間量の緩和下限、下限、上限、緩和上限を考慮できるようアークコストと流量の上下限を決める。緩和下限アークは、流量の下限を0、上限を l_i^+ とし、アークコストを絶対値の大きな負の値にする²。下限アークは、流量の下限を0、上限を $(l_i - l_i^+)$ とし、アークコストを負の値(例えば、 $-w_i^-$)とする。上限アークは、流量の下限を0、上限を $(u_i - l_i)$ とし、アークコストを0とする。緩和上限アークは、流量の下限を0、上限を $(u_i^+ - u_i)$ とし、アークコストを正の値(例えば、 w_i^+)とする。また、スーパーノードからダミーヘルパーノードへは、コスト0、流量の上下限がサービス時間の総和となるアークを設定する。

さらに、すべてのサービスノードからスーパーノードへアークを設定し、アークコストを0とするとともにアーク流量の上下限をサービス時間量 $(e_s - b_s)$ と等しく設定する。ダミーサービスノードからスーパーノードへは、コスト0、流量下限が0、上限がサービス時間の総和となるアークを設定する。

図1にネットワークのイメージを示す。左右にあるスーパーノードは、実際には同じ1つのノードを表しており、循環ネットワークとなっている。そのため、この問題は最小費用流問題となっており、訪問介護事業所程度での問題の大きさ程度であれば、瞬時に最適解を求めることができる。

提案するアルゴリズムは、この緩和問題を子問題と

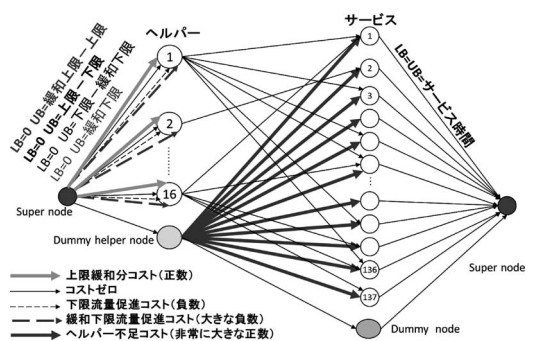


図1 最小費用流ネットワーク

して持つ分枝限定法である。上記の問題は、移動時間も含めたサービス時間の競合や、1つのサービスは1人のヘルパーが提供するという条件が緩和されているため、ダブルブッキングや1つのサービスを複数のヘルパーで時間を分けて扱うようなスケジュールを与える可能性がある。そのような場合、そこに関わっている変数を1つ選び、その変数を0および1に固定することで問題を分枝するというものである。

一方、サービスの数は1週間で数百程度、1カ月で数百~数千になるのが一般的だ。それら1つ1つに対し、各ヘルパーが担当するか否かに対応する変数が必要なため、比較的変数の数が多くなる。そこで、本研究では、問題サイズを極力小さくするために、全体最適性にもほとんど影響のない方法として、1週間ごとの部分問題に分けて問題を扱うことにした。これは、ヘルパーの勤務可能時間帯やサービスに関する登録が1週間単位であることに依存するだけでなく、週を分けて解いても、勤務時間量の緩和上下限が極端に厳しくない限り、最重要とされる「カバーされないサービス時間量の最小化」が達成されるからである³。そこで、1週間分のスケジュールを週の数だけ繰り返し作成する方法を採用した。

分枝限定法における子問題の緩和問題(最小費用流問題)が高速に解けること、問題サイズが大きくなりすぎないことから、最適解を高速に得る仕組みを準備することができた。訪問介護事業所で実際に運用した際にも、未割当てのサービスが少なく、良質な解を得ることができている。

さらに、事業所ごとの具体的な考慮点を組み入れら

² 流量の上下限とも l_i^+ に設定してアークコストを0とするほうが、定式化に沿うものの、現場においては「実行不可能」という結果を出すことが望ましくないため、流量の下限を0とする設定を採用した。そして、その解が本当の意味で実行不可能であるか否かを人間が判断できるようにした。

³ 週ごとの勤務時間量に厳しい緩和上限が設定されると、1カ月を通しての緩和上限より条件がタイトになってしまう。また、 W の値を大きく設定して「カバーされないサービス時間量を最小化」するために、緩和下限を守れない可能性も残る。

れるよう、サービスごとにヘルパーを指定できるようにし、指定ヘルパーが休みの場合にもほかの担当可能ヘルパーに自動割当するか否かも選択できるようにした。現場においては、サービス時刻の重なりを考慮して担当可能ヘルパーを決めていること、ある程度の数のサービスにヘルパー指定があることから、緩和問題の解は比較的早く実行可能スケジュールとなる。結果として、探索木も大きくはならず、一般的に分枝限定法が抱える時間的リスクも少なくなっている。

5. web システムの構築

本研究では、以上のアルゴリズムを Web システムとして実装した。Web システムとすることで、システムの存在の周知や導入の容易さをはかって運用コストを下げる意図がある。ただし、データ入力と解の修正に関しては、一般の表計算ソフトの編集機能を利用し、その完成されたインターフェースと機能を利用することで、利便性を向上させると共に開発コストを抑えた。対象問題はデータが比較的大きく、すべてのユーザ／ヘルパーの情報を一画面内に表示することが困難であるため、大きな表を扱う機能に長けた表計算ソフトの機能を利用することにしたのである。本システムの開発には多額の費用が必要であったが、これは「研究のコスト」として、運用コストに導入することは考えないことにしたい。運用コストはあくまで業務を行う当事者が払うコストであり、われわれ研究者の払うコストは別次元であると考え、研究のコストをどれだけ払うかは、研究の重要性から判断されるべきものである。

基本的な作業の流れは以下ようになる。ユーザはホームページにアクセスし、表計算ソフトのファイルをダウンロードする。ヘルパーと利用者名簿（名前情報）を作成するとともに、もう1つにヘルパーと利用者の基本情報（各曜日について登録されている情報）とスケジュールする月の情報（対象月のみのサービスの変更、勤務希望など）を入力し、アップロードする。この際、個人情報保護のため、名前を ID に変換する処理を行う。スケジューリングはサーバ上で自動で行われ、その解を表計算ソフトのファイルとしてダウンロードする。ファイルには未割り付けのサービスなどが表示されており、ユーザの考えに基づいて変更を行う。以下に、この作業の詳細を記述する。このシステムは、7章で紹介する URL 上で実際に利用できる。

図 2 は、本システムの Web 画面である。ヘルパー数と利用者数を入力することにより、スケジュール作成

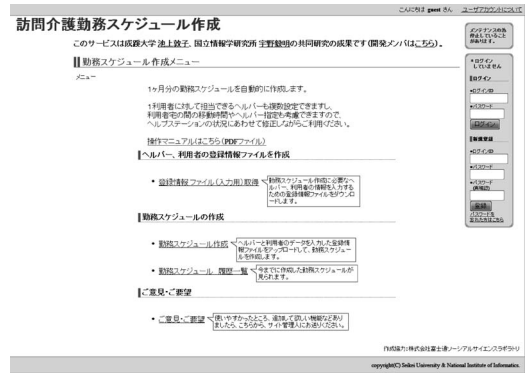


図 2 システムの Web 画面



図 3 メインメニュー画面

に必要な情報を入力するためのエクセルファイル（名簿ファイルと入力用ファイル）をダウンロードできる。

A. データ入力

名簿にヘルパーと利用者の名前を入力した後、メインメニュー（図 3）に従い、名簿に対応した ID 変換を行うことで、入力用ファイルに個々の名前とデータ入力欄が準備される。そして、入力用ファイルに対し、年間を通して比較的变化の少ない基本情報の入力を行う。以下に、それぞれの入力内容と登録シートを示す。
ヘルパー基本情報：名前、勤務可能曜日と時間帯、1週間あたりの勤務時間量の上下限、上下限が守られなかった場合に絶対守るべき上下限、ヘルパー間の勤務時間量の上下限を守る優先度。

利用者基本情報（図 4）：名前、各サービスの曜日と時刻、サービス内容、指定ヘルパーの有無、担当可能ヘルパーのリスト。

移動時間（図 5）：利用者宅間の移動時間。

以上が、システムの初回利用の際に必要な入力となるが、2回目以降の利用においては、新規登録や登録

サービス	時間	曜日	ヘルパー	備考
...

図 4 利用者基本情報登録シート

勤務スケジュールを作成する

スケジュール作成

入力済みの登録情報ファイルを選択して「サービス」タブをクリックし、スケジュールを作成します。
 印刷が可能な状態で、指定したサービスと勤務時間ファイルを選択し、下のスケジュールの印刷ボタンをクリックしてください。

スケジュール作成

1. 入力済みの登録情報ファイルの登録
 ファイル選択
2. スケジュール作成の方針
 ● 自動割付 (標準設定)
 ● 自動割付を行わず (指定ヘルパーが休みの場合は保留)
 ● 自動割付を行わず (指定ヘルパーのみ割付)
3. 最大実行時間
 ● 15分 (標準設定) ● 5分 ● 10分 ● 30分 (すべて1分単位)
 ● 実行時間なし (指定ヘルパーのみ割付)

印刷ボタンをクリックして印刷します。
 印刷されたスケジュールは、指定したサービスと勤務時間ファイルに基づいて印刷されます。
 印刷されたスケジュールは、指定したサービスと勤務時間ファイルに基づいて印刷されます。

図 7 web 画面 (アップロード)

ロードする (図 7)。個人名が流出しないよう ID 変換しないとアップロードできないようになっている。Web 画面でスケジュール作成方針と実行時間の上限を選択する。作成方針は自動割付が基本だが、指定ヘルパーが休みの場合は、他の担当可能ヘルパーを自動割付する場合 (標準設定) と、休みの場合は保留とする場合の 2 つから選べる。さらに、自動割付を行わずに指定ヘルパーのみ割り付けたスケジュールを与えることもできる。

データ入力後 (もしくは、データ入力中に) 入力データの整合性のチェックを行う。移動時間を含め、競合するサービスを同じヘルパーに指定していないかなどをチェックできる。

実行時間の上限は、1分、5分、10分、30分から選択できるが、設定上限より早く最適解 (モデルに対する最適解) が得られれば、作成された勤務スケジュールが自動でダウンロードされる。時間内に得られなかった場合は、暫定解がダウンロードされるが、われわれの調査では、ほとんどの場合、数秒から数十秒で最適解が自動ダウンロードされている (計算時間は数秒)。

ダウンロード後、ID 変換することにより、全員分の実名の入ったスケジュール (全体勤務表) を得る。

C. 編集 (修正, 印刷)

作成された全体勤務表に対して直接、もしくは、1日ごとのガントチャート (図 8) 上で修正を行う。最終的な勤務表が完成したら、個々のヘルパーの勤務表 (図 9) と、利用者のサービス提供表 (図 10) を印刷して配布する。

利用者のサービス提供表の文字はできる限り大きくなるようフォーマットを工夫した。

利用者名	サービス	時間	曜日	ヘルパー	備考
...

図 5 移動時間変更シート

月	火	水	木	金	土	日
...

図 6 ヘルパー月間情報シート (1 人分)

内容の変更のみ行い、毎月行う入力と作業は、以下のものとなる。

月間情報登録: 基本登録情報に対し、対象月における休み希望やサービスの追加などの予定変更をヘルパーごと (図 6)、利用者ごとに入力する。

B. スケジューリング

スケジュール作成アルゴリズムを利用するために、データ入力したファイルを ID 変換し、Web にアップ

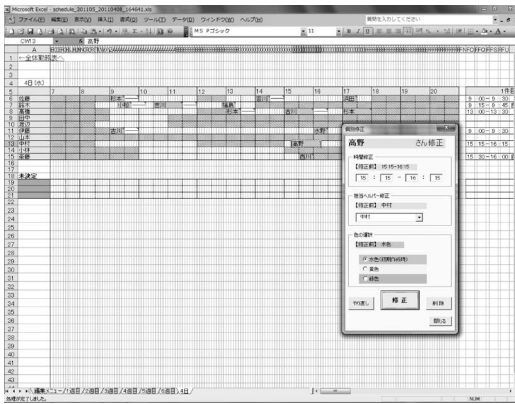


図8 ガントチャート (1日分)

月	火	水	木	金	土	日
9月	10日	11日	12日	13日	14日	15日
9月	16日	17日	18日	19日	20日	21日
9月	22日	23日	24日	25日	26日	27日
9月	28日	29日	30日	10月	1日	2日
10月	3日	4日	5日	6日	7日	8日
10月	9日	10日	11日	12日	13日	14日
10月	15日	16日	17日	18日	19日	20日
10月	21日	22日	23日	24日	25日	26日
10月	27日	28日	29日	30日	31日	

図9 ヘルパー勤務表 (1人分)

月	火	水	木	金	土	日
9月	9日	10日	11日	12日	13日	14日
9月	15日	16日	17日	18日	19日	20日
9月	21日	22日	23日	24日	25日	26日
9月	27日	28日	29日	30日	10月	1日
10月	2日	3日	4日	5日	6日	7日
10月	8日	9日	10日	11日	12日	13日
10月	14日	15日	16日	17日	18日	19日
10月	20日	21日	22日	23日	24日	25日
10月	26日	27日	28日	29日	30日	31日

図10 利用者サービス提供表 (1人分)

6. システムの評価

この節では、前節で紹介した Web システムを、小規模ビジネス OR としての観点を中心に、主観的ではあるが評価したい。まず、基本的なモデル構築であるが、これは多数の事業所へのアンケートを基に構築したものであり、多くの事業所に通じる共通な構造を抽出したといっていよう。ただし、アンケートには現れ

ていない、隠れた構造が存在する可能性も否定できず、今後のさらなる調査、システム利用の履歴やユーザからのリクエストといったところから、今まで見ていなかった構造を探り出すことが課題となる。また、モデル自体はシンプルであり、設定の難しい複雑な要因を含まない。これはモデルの柔軟性という点で評価できると考えている。

最適化以外、可視化などの機能については表計算ソフトのマクロで実装した。これにより、非常に軽く機敏に操作でき、かつ強い機能を持つシステムを実現することができた。また、過去の計算結果、操作履歴を記録する機能を実装し、将来的に問題をより深く分析するためのデータ収集を行えるようにした。また、機能に関するリクエストを送付できる機能を盛り込み、ユーザの要望が収集しやすいようにした。以下、運用コストに関わる点を中心に項目ごとに評価を行う。

導入: Web システムを用い、無償で利用提供している。また、導入の手間も不要である。必要な経費は、コンピュータとインターネットへの接続費用であるが、この2つはたいの事業所はすでに備えていると考えていだろう。操作方法の習得であるが、Web のメニューは直観的にわかりやすいよう、レイアウトとメッセージの文言に気を遣っている。実地調査では、現場のスタッフに何の説明をせずとも、画面を見るだけで操作が理解できていた。また、データ入力に一般的な表計算ソフトを使用しており、この操作もたいの習熟しているスタッフがいるため、コストは低い。表計算ソフトの使用経験のないユーザでも、既存の豊富な解説書を参照できるため、敷居が低いと考えられる。ほかに、データの名前などを ID 変換してからサーバにアップロードするようになっており、個人情報自動的に保護されるようになっている。これにより個人情報漏洩の心配から IT システムの利用を敬遠するユーザにもある程度の安心感を与えられると考えられる。実際、今回の実験でも具体的な操作方法についてはそれほど説明せずとも理解された。

データ入力: 基本情報と月次情報にファイルを分けたことで、不要な二重入力を回避し、手間を削減している。また、表計算ソフトの良くデザインされたインタフェースを利用しているため、ユーザへの負荷を軽減できている。また、ファイル単位で管理しているため、ファイルのコピーなどを用いて、データの管理やコピーと言った作業が容易になっている。欠点は、表計算ソフトと Web サービスの両者を利用する必要があり煩雑である点と、データの受け渡しや ID 変換などを全自

動化することができず、ユーザに2つほど、ある種無駄な操作を要求している点である。モデルで使うデータに関しては、利用者宅間の移動時間などにデフォルト値を導入し、デフォルト値の利用を前提としたシステム設計を行うことで多大なデータ入力を軽減することに成功している。

連絡: 利用者・ヘルパーからの情報入力については、本システムでは何の機能も持っていない。ただし、利用者・ヘルパーへの連絡については、各個人用の1カ月のカレンダーを自動作成する機能を有するため、迅速かつ手間の少ない伝達が可能である。また、ユーザによる印刷フォーマットのカスタマイズ機能により、余白に情報を書き込む、高齢者でも見やすい印刷物にする、といった作業がやりやすくなっており、この点でも連絡の手間の軽減を達成している。結果として、手書き作成に比べ、大幅な時間短縮を達成できている。

勤務の質: 定式化に、勤務時間の上下限を2種類入れることにより、ある程度公平性及び適正バランスを担保するようなスケジュールの実現を行っている。最適化は、コストの安い人物・部署に仕事を集中させるという癖があるため、この制約がないと非常に偏った解が発生しやすく、この制約の果たす役割は大きいと考える。また、以下に示すように、充実したインタフェースを持つため、突然の予定変更など、運用中の計画変更も比較的容易にできるようになっている。

計画: 求解にかかる計算コストは非常に小さく、ヘルパー・利用者200名程度の規模であれば、通常パソコンからのサーバ送受信も含め数秒以内に求解できる。実地調査では、未割当てなど制約違反も比較的少なく、修正にかかる手数は比較的小さい。求まった解は表計算ソフトのシートとして表現されるため、閲覧性が高い。また、違反した制約や未割当てのサービスを目立つように色づけたことで、修正が必要なポイントを上手に可視化することに成功している。解の修正は表計算ソフトの豊富なインターフェース機能を用いてできるため、操作性が非常に高く、制約違反の可視化を更新する時間も短いため、この点でストレスを感じることも少ないと思われる。また、月間情報の入力中に基本情報の変更を思い出した場合（その月から基本情報自体も変更される場合）にも、それらを一緒に扱えるよう、月間情報登録画面から、それらに関わる基本情報が変更できるようにした。これにより画面の移動時間が飛躍的に短くなり、入力に関わる操作性が向上した。われわれの実地調査では、以前は毎月数十時間かけていた作業が、ほんの少しの習熟時間と初期設定

に加え、わずか2時間程度でできるようになり、システム導入による運用コスト削減が劇的に行えることが確認できた。また、過去の作業では多発していた割当てミスが、本システムの利用後は発生しなくなり、この点でも大きな作業効率の改善があった。

透明性: 定式化が定まっていること、基本的な制約しか考慮していないことにより、求まった解が何の意味で最適であるのか、容易に理解できる。この意味で、求まった解には透明性があり、納得感もあると考えられる。また、複雑なパラメータ設定など、ある種の職人芸は必要ないため、引継も容易である。システム自体、過去の履歴を保存する仕様になっており、過去のデータを引き継いだ最適化が可能である。これも、調査中に観察されたことだが、勤務時間量の過不足がその上下限值設定から起きたことを自ら理解し、当初の緩い制約のデフォルト値を設定しなおして求解を繰り返すことが行われていた。移動時間の設定についても同様なことが観察された。

創意工夫: 担当ヘルパーの対応表や全体スケジュールを1枚のシートにすることで、俯瞰しやすい可視化を達成しており、違反した制約の色づけで、修正点が必要な点、制約が厳しい点を把握しやすい。また、ガントチャートや個別スケジュールなど、最適化エンジンには不要だが、修正をする際に助けになるような可視化ツールを複数用意することで、解の構造を理解しやすくした。編集結果を反映するのが速いので、ユーザの思考を妨げない。これらにより、ユーザが創意工夫を行いやすい環境ができていると言ってよいだろう。

7. おわりに

2011年4月に、2つのヘルプステーションにおいて、本支援システムを利用した勤務スケジュール作成が行われた。初めてシステムを利用したヘルプステーションでも、マニュアルを利用しながら基本情報入力から全体勤務表までスムーズに作成することができた。一方、利用経験のあるヘルプステーションにおいては、マニュアルなしでも迷うことなく全体勤務表まで作成することができた（ヘルパー35名、利用者104名に対して、頻繁な他業務割込みの時間も含めて、2時間弱）。月末までに入ってくる追加の変更情報も加え、ガントチャートを利用した修正を行っているが、急なサービスの追加に対しても、ガントチャート上で挿入の可能性を確認できるため、ミスのないスケジュールを完成できると報告された。

解空間把握のための情報をどのような形で提供する

かについては、ガントチャートのみに留まったが、今後は、この情報のあり方に加え、「実用上の最適解」を把握しやすい解、つまり修正しやすい解がどのようなものであるかについて研究を進めたい。

本研究のシステムは、2011年5月から無料で公開されている (<http://homehelp.nii.ac.jp/>) 本システムが介護事務所の作業軽減に資することを願うとともに、今後もより良いシステムへと改良を続けていきたい。

謝辞 本論文に貴重なご助言を頂いた査読委員の先生方に感謝いたします。また、本研究に多大なるご協力をいただいた至誠学舎立川至誠ホームの皆様、システム作成に尽力いただいた株式会社富士通ソーシャルサイエンスラボラトリの皆様に感謝いたします。本研究の一部は、文部科学省私立大学戦略的基盤形成支援事業研究費の成果、政策研究大学院大学の客員研究員としての成果を含んでいる。

参考文献

[1] A. Ikegami, A. Niwa, “A Subproblem-centric Model

and Approach to the Nurse Scheduling Problem,” *Mathematical Programming*, **97**, 517–541, 2003.

[2] A. Schaerf, “A Survey of Automated Timetabling,” *Artificial Intelligence Review*, **13**, 87–127, 1999.

[3] E. K. Burke, P. De Causmaecker, G. V. Berghe, H. V. Landeghem, “The State of the Art of Nurse Rostering,” *Journal of Scheduling*, **7**, 441–499, 2004.

[4] 池上敦子, 緒方洋平, 森田隼史, 土谷 隆, “訪問介護スタッフ・スケジューリング,” 統計数理研究所共同研究レポート 191 「最適化：モデリングとアルゴリズム 19」, 302–316, 2006.

[5] 吉田勇人, 軍司奈緒, 池上敦子, “訪問介護勤務表作成の現状と作成支援システム,” 経営情報学会秋季全国研究発表大会, セッション ID : A2-1, 2011.

[6] 久保琢磨, 宇野毅明, “中小規模スタッフスケジューリング問題における調整の容易なスケジュール作成に関する研究,” 情報処理学会研究報告, 2008-MPS-65, 57–60, 2008.

[7] 厚生労働省, “平成 22 年介護サービス施設・事業所調査結果の概況,” <http://www.mhlw.go.jp/toukei/saikin/hw/kaigo/service10/index.html>

[8] A. Ikegami, A. Uno, “Bounds for Staff Size in Home Help Staff Scheduling,” *Journal of the Operations Research Society of Japan*, **50**, 563–575, 2007.

[9] P. Egeborn, M. Rönnqvist, “Scheduler-A System for Staff Planning,” *Annals of Operations Research*, **128**, 21–45, 2004.