

# 実用性を重視した組合せ最適化技術の応用 —多様な解の出力と計算資源制約—

田中 俊明, 半田 恵一, 愛須 英之, 山田 尚史, 國信 茂太

最適化, スケジューリング技術が実問題に適用される場面が増大している。求めた解がどのように生かされるのかを考えると最適性や高速性以外にも様々な解決しなければならない課題が現実には存在する。本稿では、その中から多様な解候補を出力することと組込み系に必ず発生する厳しい計算資源制約について実際に開発してきた事例をもとに紹介する。

キーワード：乗換え案内, 動的資源割当て, エレベーター群管理, PC 筐体設計, 多様性, 資源制約

## 1. はじめに

最適化, スケジューリング技術は, 社会に幅広く応用されている。最近では, データの電子化が一層進んだこと, システムが複雑になり熟練者でも短時間での処理が困難になってきたことなどにより計算機による支援が増えてきている。通常, 実際の問題は最適化やスケジューリング問題として与えられるわけではないので対象のモデリングが重要になる。熟練者がやっている場合には, 現場に出かけて実際にどんな知識を使って問題解決を図っているか, 問題の規模など現状の課題のヒアリングから始まる。そして最終的にどのような使い方になるのか, どのような計算資源を使って実装するのも考えていかなければならない。実用化では利用形態から迫られる大きな課題がある。モデリングは最も重要な第一段階である[1]。本稿では筆者らのこれまでの実用化研究の中から重要だと考えている課題について述べる。

第一の課題は, 多様な準最適解の出力である。計算機が出した解を意思決定の判断材料とする場合には, 最適解より多様な準最適解を出力することの方が重要なことが多い。あくまで人間の意思決定を支援する機能だからである。例えば, 後で紹介する乗換え案内における複数経路の提示はその典型である。最適経路として一つだけを提示しても不満が残るだろう。最適経路の明確な基準が定めにくいということ, 決めるの

は利用者自身であるということが理由である。したがって, いくつかの経路から利用者が選択するのがよい方法といえる。これまで熟練者がやっていたことを代替する場合にはこのタイプの課題がよく発生する。これに対して人間が介入しない場合には複数の解は意味がない。計算機が出した解で次の処理に移ることになるので一つの解でなければならない。

第二の課題は, 計算資源制約である。最近では組込み系も大規模化が進み, 最適化機能が重要になってきている。しかし, 組込み系では使える計算資源 (CPU, メモリ等) の厳しい制約がある。したがって, その制約の中で最適化を実現しなければならない。高速解法以外にも, メモリ制約や途中で処理が打ち切られてもそれなりの解を出力できるロバスト性などが重要になる。

第一の課題の例として乗換え案内エンジン, PC 構想設計支援技術を取り上げる。乗換え案内は既に(株)駅探でサービスされているものであり, PC 構想設計支援は PC の筐体設計者を支援する最適化機能として開発しているものである。

第二の課題の例として, デジタル機器の動的資源割当て, エレベーター群管理制御方式を取り上げる。デジタル機器の動的資源割当ては当社のハイビジョンレコーダーの番組録画機能として搭載されているものであり, 群管理制御は次世代方式として開発中のものである。

## 2. 乗換え案内エンジン

列車や飛行機等の乗換え案内サービスにおける経路探索には実用上の様々な制約条件が伴う。例えば, 乗換えや待ち時間, 各種の列車や飛行機の混在, 他線へ

たなか としあき, はんだ けいいち, あいす ひでゆき,  
やまだ ひさし, くにのぶ しげた  
(株)東芝 研究開発センター  
〒212-8582 川崎市幸区小向東芝町1

の乗入れなど。また経路の良さにも、所要時間、乗換え回数、料金などユーザや利用目的によって様々な評価基準がある。筆者らはこれらの制約条件や評価基準を考慮に入れた上で多様な複数の経路を高速に検索するエンジンを開発し、駅探の乗換え案内サービスに実装した[2]。

当エンジンは、経路探索、時間割当、料金計算という一連の機能からなる。ユーザにより乗車駅および降車駅、出発または到着時刻などの検索条件が与えられると、まず経路探索によって多数の（例えば数百本の）経路が候補として列挙される。次に時間割当によって各経路候補に対して具体的な列車が割当てられる。すなわち、乗降車駅と乗換え駅における発着時刻が割当てられる。最後に、料金計算によって各経路候補の料金が計算される。経路探索によって列挙された候補は、時間割当および料金計算のプロセスにおいて徐々に取捨選択され最終的に数本の経路がユーザに案内される。

経路探索のベースとなるのは第1~K最短経路を求めるアルゴリズムである。筆者らは、多様な経路を求めるためにネットワークと探索手法に様々な工夫を行っている。

複数の経路を求める場合、MPS法[3]等のアルゴリズムをそのまま適用したのでは実用的に不十分である。経路が平均所要時間だけで評価され、しかも類似の経路が多数選択されるからである。乗換え回数や料金も重要な評価基準である。平均所要時間の短い経路であっても、実際に時間割当をしてみたら列車の接続が悪く所要時間が余計にかかるケースやその逆のケースがある。本手法は、経路探索、時間割当、料金計算の順に処理が進むため、経路探索の最初の段階ではなるべく多様性に富んだ経路群を求めておく必要がある。

路線ネットワークは、図1のように、路線図をベースに、駅をノード、路線と乗換えや隣接する駅への徒歩移動をアークで表現する。山手線と京浜東北線のように一部並走する路線があると、それらの路線間での様々な乗換えパターンが可能となり、その結果、互いに似通った経路が頻繁に出力される可能性がある。それを避けるために乗換えアークの一部を前もって削除している。多様な経路を求めるということは、別の言い方をすると類似経路を抑制することでもある。

また、あらかじめアーク群を路線の種類に応じてグループ分けし、飛行機路線を含まないネットワーク、さらに新幹線を含まないネットワーク等に順次切り替えることによって多様な経路を求めている。平均所要

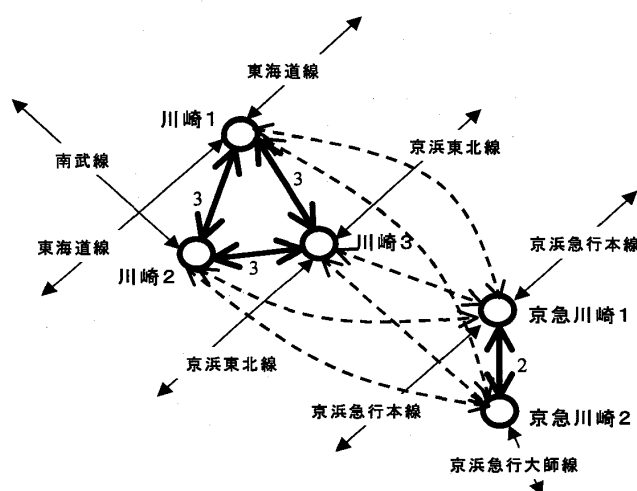


図1 路線ネットワークのモデル (川崎駅)

時間の短い経路から順に探索するため、遠距離の検索を行うと、例えば、飛行機や新幹線を利用した経路ばかり求まることもあるからである。乗車駅と降車駅が遠く離れていても別料金が不要な快速や各駅停車で行く経路や在来線特急だけで行くような経路も欲しい。

経路の多様性を高めるためのもう一つの工夫として、次のような通常の経路探索後の再探索を行っている。

#### (1) 乗換えの少ない経路の再探索

徒歩および乗換えアークのすべてにある一定時間を一時的に加算して上位最短経路をいくつか求める。ただし、乗/降車駅内の乗換えアークや乗入れのある乗換えアークには加算しない。

#### (2) 使用頻度の高いアークを避けて再探索

通常の経路探索と再探索(1)で求めた経路群において使用頻度の高いアークの重みを大きくしてそのアークを使用しない他の経路を求めやすくして上位最短経路をいくつか求める。

例えば、立川から新潟への経路として立川—中央線—東京—新幹線とき—新潟のような経路がある。通常の探索では普通列車を乗り継いで大宮まで行き、そこから新幹線を利用する経路が求まりやすいが、再探索(1)によって東京から始発の新幹線に乗るといった経路を案内することも可能となる。

以上のように様々な工夫で多様な経路を出力している。

### 3. PC 構想設計支援技術

PC筐体設計では、ノートPCの筐体内にCPU・ハードディスク・USB端子等、比較的サイズの大きい主要部品をどのように配置するかを決定する。最近のノートPCは、高機能・多機能化により、配置する部

品数が増えてきており、コンパクトな配置設計をすることが困難になってきている。また、PC 部品のもつ様々な配置制約が問題をさらに難しくしている。設計された配置案は、次の工程で熱解析・強度解析等に行われ、それらの検査をパスしない場合、別の設計案を作成する必要がある。PC 筐体設計の構想設計初期作業を支援するのが PC 構想設計支援技術である。開発した PC 構想設計支援ツール[4]は、筐体内に配置する部品名・配置制約・筐体の縦横サイズ等を入力とし、厚みの薄い複数の配置案を出力する。複数の配置案を出力することにより、ある配置案が熱解析・強度解析等の検査をパスしない場合でも、設計者は代替案を利用することができる。

部品を筐体内に配置する問題は、3次元ビンパッキング問題の応用と考えることができるが、部品に配置制約がある点、多様な解を求める点（最も厚みの薄い配置案が最良とは限らない点）が大きく異なっている。開発した PC 構想設計支援ツールが扱える部品配置制約の代表的なものは、配置層制約と水平配置位置制約である。配置層制約は部品の鉛直方向の配置位置に関する制約である。例えば、キーボードは筐体上面に配置、CPU は基板上面に配置、メモリは基板下面に配置等が配置層制約となる。一方、水平配置位置制約は、水平方向の配置位置に関する制約で、部品ごとに定められている配置範囲内に配置しなければならないという制約である。例えば、電源端子は背面に接する位置に配置範囲が定められている。

部品配置案の算出は、配置層（2次元平面）ごとに部品を配置した後、3次元に集積することによって行われる（図2参照：筐体内部層は、筐体内であればどこでも配置可能な、他の層に属さない部品が配置される層）。配置層上で配置位置を決定する際には、既に配置層に配置されている部品の位置およびその時点での筐体厚み等を考慮する。

多様な部品配置案を出すための工夫としては、配置する部品種類のランダム選択および部品の配置順の確率的決定がある。PC 部品の場合、一般に、同一機能を有する複数の部品種類が存在する。例えば USB 端子には、縦型・横型・2連型等様々な種類が存在する。PC 構想設計支援ツールは、同一機能を有する複数の部品種類が存在する場合、それらの中からランダムに配置する部品を選択する。また、一般に、部品の配置順が異なれば、異なる配置案が算出されることから、配置順の決定には確率的要素を含めている。配置順は

ランダムに決定しても多様な配置案を算出することは可能だが、部品サイズ（水平投影面積）の小さいものを最初に配置してしまうと、後々、サイズの大きな部品を配置する空間が確保できなくなり、配置案そのものが算出されにくくなる。そこで、計算時間と解の多様性とのトレードオフを考え、部品 A（サイズ  $a$ ）と部品 B（サイズ  $b$ ）では、 $\{a/(a+b)\} \times 100 [\%]$  の確率で部品 A が先に部品 B より先に配置されるように配置順を確率的に決定している。

PC 構想設計支援ツールには、多様な解を高速に算出する工夫以外にも、現実的な配置案を算出するため様々な工夫が含まれている。

本ツールは、部品数が 57 のノート PC 設計の場合、約 5 分で 30 弱の配置案を出力する（Windows XP Core 2 Duo 266 GHz 2 GB RAM Java Version 1.6.0 update 7）。また、PC 設計者の設計した配置案と同様の配置案も算出されることも確認されている。図3は PC 筐体設計支援ツールが出力した配置案の例である。

#### 4. デジタル機器の動的資源割当て

近年、多チャンネル化が進み、多数の放送を同時録画し大量に自動録画・蓄積するスタイルが主流になり、さらに、家庭用録画機器の多機能化により複数の録画

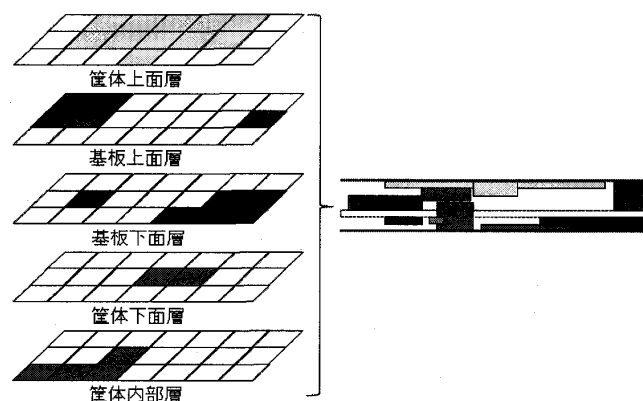


図2 部品配置

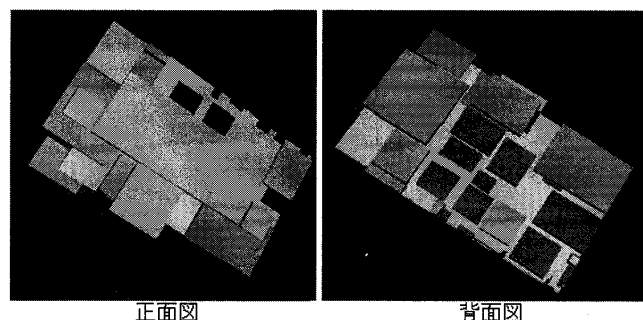


図3 算出された配置案の一例

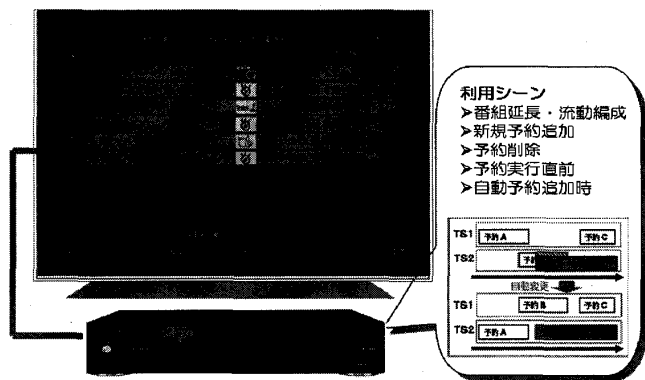


図4 録画予約への資源割当て

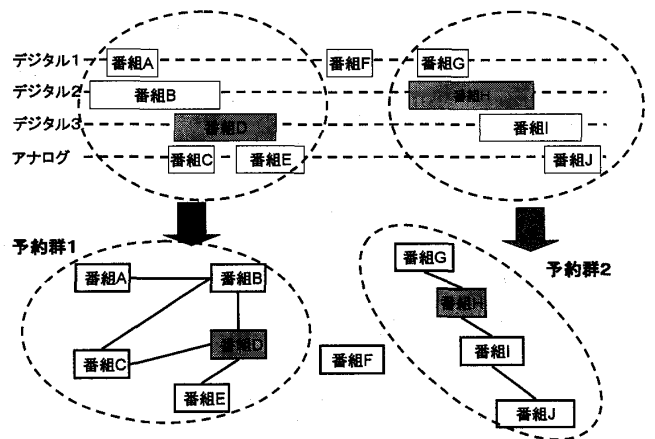


図6 予約のグラフ表現と予約群の抽出

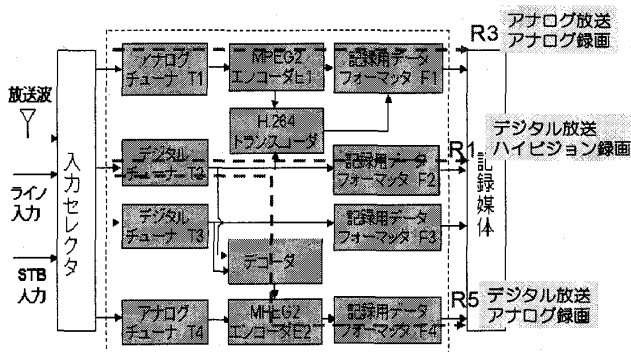


図5 モジュール構成の例

資源を様々な録画モードに応じて選択することが一般化している。このため、ハイビジョンレコーダー等の録画機器向けに、録画予約の登録削除や放送時間変更の際に時間帯が競合する各予約にチューナやエンコーダ、トランスコーダなどの共有資源を最適に振り分けるタスクスケジューリング機能を開発した[5](図4)。

本問題は、汎用計算機に比べ資源の競合関係が複雑なため簡易なタスクスケジューリングが難しい。家電機器で組み込みリアルタイム処理するために、演算負荷とメモリ使用量を抑える必要があり、問題をグラフ彩色問題として一般化し、複雑な資源の排他関係を汎用的に扱えるようにモデル化を行っている。

録画方法は録画モードに応じ、複数のモジュールを組合せて使われるので、そのモジュールの組合せを仮想的な一つの資源として定義し、これを各予約に割当てる問題としてモデル化している。モジュールの共有関係に応じて、それぞれの仮想的な資源の間の排他関係を定義する。例えば、図5のモジュール構成の例では、アナログ放送アナログ録画(R3)を指定した予約に対しては図の点線上にある処理モジュール(資源)の組合せで信号を処理しなければならない。R1とR3の録画モードは資源を共有しないので同時録画

可能であるが、R1とR5はデジタルチューナーT2を共有するので同時録画できない排他関係にある。

次に、予約番組の間の競合関係をグラフとして表現する。図6は予約番組を時間軸に沿って並べ、対応する部分をグラフとして表現した例である。ここで、各ノードは予約番組を示し、アークは予約番組間で録画時間帯が重複していることを表現している。

図4の利用シーンにあるように、新たに予約が追加・削除、時間変更された場合に割当/再割当演算が実行される。その発端となった予約を基点予約とし、直接、間接に録画時間帯が重複する予約だけを抽出し、グラフを作成する。これで割当演算範囲を必要最小限に抑えている。

予約への資源割当は、グラフ彩色における色の割当に相当する。各ノード(予約)に割当て可能な色候補(資源)はノードによって異なるのでリスト彩色問題と見なせる。通常のリスト彩色問題との大きな違いは、隣のノードに資源が割当てられると、その資源だけでなく排他関係にあるすべての資源が使えなくなることである。提案手法では、アークの多いノードはそれだけ競合が多く制約が厳しいので、先に彩色するように彩色順番を決めている。

本アルゴリズムは、高々  $O(N^2)$  ( $N$ はノード数)の演算時間であり、グラフ表現はその場で生成するため記憶しておく必要がなくメモリ使用量も少なくて済む。

本技術は、現在、当社のVARDIA™シリーズにおいて「W録自動振替機能」として搭載されている。将来は、複数の録画機器間で相互に資源を融通するような高度連携機能への展開も検討している。

## 5. エレベーター群管理制御方式

エレベーター群管理制御とは、乗客へのサービス面

(待ち時間など)とビル管理者の運用面(省電力など)の両面から効率的に運行させるために、乗り場階に発生するホール呼びを適切な号機に割当て決定するための技術である。従来からこの群管理制御にはAI、ニューラルネット、GA等の技術が応用されてきたが、近年マイクロコンピュータの性能が飛躍的に向上し、高機能な計算資源を利用してより複雑な演算が行えるようになってきた。そこで当社では、ホール呼びが発生するごとに将来にわたる交通需要を考慮した運行スケジュールリングを実行し最適な号機割当てを行う次世代の群管理制御方式を開発している[6][7]。

次世代群管理制御方式の特長は以下の通りである。

第一の特長は、現在だけでなく将来を考慮していることである。高頻度に発生することが統計的に分かっている階間交通需要を将来のホール呼びという形で複数個発生させて、これらホール呼びにตอบสนองすることも併せた運行スケジュールリングを行うことで、将来を予測した号機割当てを行っている。

第二の特長は、組合せ最適化問題として運行スケジュールリングを扱っていることである。ホール呼びを号機に割当てると運行スケジュールは一意に決まるので、平均待ち時間等が評価できる。したがって、将来のホール呼びも合わせた複数個ホール呼びの運行スケジュールリング問題は、各ホール呼びを適切な号機に割当てる組合せ最適化問題に帰着できる。

第三の特長は、リアルタイム性を重視していることである。一般にシステムは、乗客の配車要求(通常はホール呼び釦の押下)から配車する号機を決定して乗客へ到着号機を報知するまでの時間が限られていることが多い。そのため、複数個ホール呼びの号機割当て組合せをすべて列挙し最適解を求めることはできない。

この課題に対し、号機割当て問題の解法としてA\*アルゴリズム[8]を応用した優良探索手法を用いている。

図7は運行スケジュールリングの探索の概念図である。探索は現在時刻( $T$ )の直前に既発生未応答呼びに対して決定していたスケジュール(初期運行スケジュール)を探索の基点とし、複数個のホール呼び( $H$ )の中から発生時刻順にいずれかの号機へ割当てた運行スケジュールを探索木のノードとして作成していく。展開済みのノードの中で最優良解を優先的に展開していくことで準最適解を導出することができる。また、限られた計算時間内における探索演算打ち切りによる割当て解の質低下を抑制している。図8はノード展開数

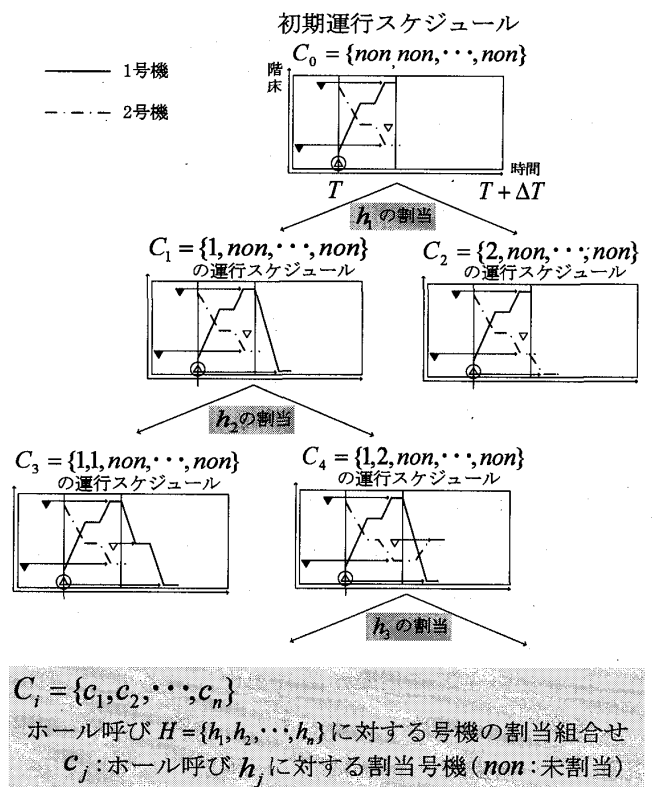


図7 運行スケジュールリングの探索概念図

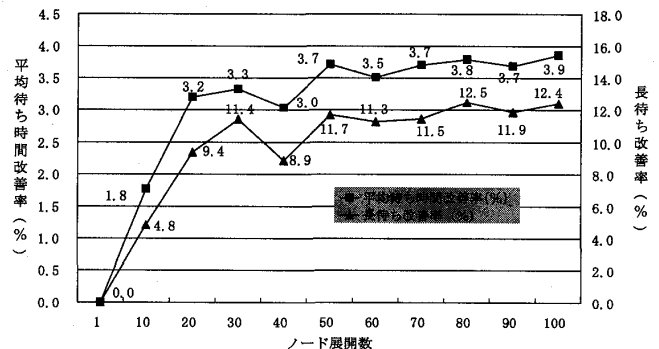


図8 ノード展開数に対する評価性能

に対して平均待ち時間と長待ち改善率(最大待ち時間)の影響をシミュレーションで調べたものである。およそノード展開数は20で飽和してくるので、この先ではどこで打ち切られてもさほど影響がないことになる。

以上述べた次世代群管理制御方式は、平均待ち時間や長待ち抑制のための割当て変更回数を大きく削減でき、団子運転の防止などにも効果が期待できる。

## 6. 考察

多様な解を出力する理由は、最初に述べたように意思決定をするのはユーザであること以外に、

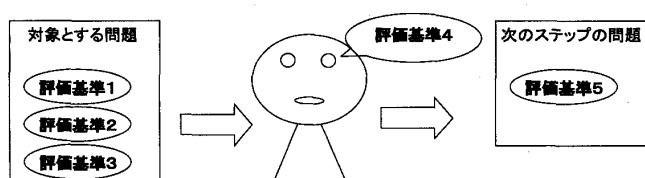


図9 評価基準のある所

- (1) 複数の評価基準があるのでバランスを保つ
- (2) 評価基準が明確に洗い出されていない
- (3) 問題を分割するために切り離す

等が挙げられる。

これを評価基準がどこにあるかで考えると、図9のようになる。上記の(1)は対象とする問題の中にあり、(2)は人間の中に漠然とあり、(3)は次のステップの問題にある。乗換え案内の事例では、所要時間、乗換え回数、料金という評価基準が問題の中にあるので、(1)として多様な解を求めている。また、乗換えが楽な経路、混雑しない経路などは(2)に属し、漠然としているが、これを対象とする問題に含めることは技術的にはある程度は可能でもシステムとしては曖昧になる。そこで多様な解を提示することでユーザーの判断に任せている。PC 構想設計では、熱・強度に強い配置を制約条件に入れることはしないで、(3)として次のステップの問題として切り離している。そのかわりに多様な配置案を求めている。部品の結合関係や配置位置の制約と熱・強度解析はレベルの異なる問題であるから同じ問題として扱うのは得策ではない。

人間の頭の中にある漠然とした評価基準は、人間が長年の経験から獲得してきたノウハウである。通常、これらのノウハウは陽に表現できない部分を含むことが多い。出力結果を熟練者に判断してもらい評価基準や優先度を決めていく学習があれば、現実に沿ったモデルになる可能性がある。

計算機や技術が進歩し、最適化、スケジューリングがリアルタイムに結果を出せるようになって人間が直接結果を利用するように応用範囲が広がっている。本稿であげた事例はすべてそのようなものである。そうになると今度は使う側の要求が厳しくなってくる。応答速度はもちろんであるが、ユーザから見て不合理と判断されるものはたとえそれが最適であっても受け入れにくいものとなる。

例えば、エレベーター群管理ではホール呼びに対して割当てられた号機を知らせる予報灯が点灯する。一度点灯するとその号機の前で待つことになるので、他

の号機が最適であってもあまり変更はしない方がよい。ハイビジョンレコーダーの資源割当てでは、ユーザが一部の予約の録画モードや優先度を指定することができる。その制約の中で最適な録画方法を決めるので、ユーザ指定がなければもっと多く録画できたとしても採用していない。あくまでユーザ優先としている。

熟練者の評価基準、ユーザの判断基準をどこまで考慮できるかは今後重要になるとと思われる。

## 7. おわりに

これまで開発してきた事例で多様な解を出力する工夫と組み込み系特有の計算資源制約を考慮した手法について紹介した。また、最適化、スケジューリング技術の適用範囲が広がるにつれて、人間との接点が増えるとその曖昧な制約条件、評価基準にどのように対応するかが課題になることを述べた。今後、これらに対応するために最適化、スケジューリング技術とデータマイニングや学習、予測、サービスサイエンスといった技術の融合もあると考えている。

## 参考文献

- [1] 久保幹雄,「モデリングのための覚え書き」,オペレーションズ・リサーチ, Vol. 50, No. 4, pp. 255-258, 2005.
- [2] 半田恵一, 田中俊明,「乗換え案内サービスにおける経路探索手法」, 電子情報通信学会論文誌, Vol. J 88-D-I, No. 10, 1525-1533, 2005.
- [3] E. A. V. Martins, M. M. B. Pascoal and J. L. E. Santos, "An algorithm for ranking loopless paths," Research Report Univ de Coimbra, 1999.
- [4] S. Kuninobu, K. Handa, Y. Sasaki and T. Iikubo, "A Parts Arrangement Algorithm for Note PC," 23th Int. Conf. on Circuits/Systems Computers and Communications (ITC-CSCC), pp. 329-332, 2008.
- [5] 愛須英之,「AV 機器の録画予約スケジューリング」, スケジューリング・シンポジウム 2008, pp. 133-138, スケジューリング学会, 2008,
- [6] 田中和弘, 高崎一彦, 山田尚史,「超大型・超高速昇降機の更なる展開」, 東芝レビュー, Vol. 62, No. 5, pp. 27-31, 2007.
- [7] 山田尚史, 愛須英之, "A アルゴリズムを用いた群管理制御の提案と性能評価," 電子情報通信学会技術研究報告, Vol. 107, No. 471, pp. 43-48, 2008.
- [8] P. E. Hart, N. J. Nilsson and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," IEEE Transactions on Systems Science and Cybernetics SSC4(2): pp. 100-107, 1968.