線形計画法の支援によるルールベース型作業スケジュール方式†

栗原謙三†† 原敬市†††
小林隆†† 汐見龍徳††††

生産，流通等におけるスケジューリング，割当てなどの計画業務においては，計画条件を事前にすべて決定することが難しい．計画条件には曖昧なものもあり，スケジュール結果によって，条件が緩和されることも多い．そのため，このような計画業務の計算機化は困難であり，従来は人手で行われることが多い．計画問題は，一般に，組合せ問題となるため，これを人手で行うことは，スケジュール担当者の負担が非常に大きい．そのため，計画立案には長時間を要し，状況変化に対する即応性の面でも問題があった．条件変更に対する柔軟性を確保し，しかも，計画の最適化も実現するために，知識工学手法と数理計画手法とを併用したスケジューリング方式を提案する．本方式では，割付け計算用に数種の最適計算アルゴリズムを用意すると共に，割付け戦略を設定するためのノウハウを知識ベース化した．そして，割付けを実施していく過程で，知識ベースにより，部分問題の抽出とこれを解くのに適したアルゴリズムの選択を行う．鉄道，航空機など，運行スケジュールが決まっていて，運転手の割付けを計画する場合を一般化したモデルを用いて，提案方式の特性を評価し，狙いどおりの効果があることを確認した．

1. まえがき

従来，計画業務の計算機システム化においては，計画問題を組合せ最適化問題として定式化し，整数計画法や分枝限定法あるいはこれらの近似解法などの数理計画法を使って解くというアプローチが多くとられている[1)~3)]．しかし，実用システムという意味では，失敗に終わることも多かった．その主な原因は，次の二点にあると思われる．

①スケジュール問題を数理計画問題に定式化すれば，その定式化された問題の最適解は得られる．しかし，近似モデルによる解法であるため，定式化しきれない部分が残り，必ずしも，実用的な解を得られない．

②計画条件変更へのフレキシビリティが得られない．すなわち，スケジュール問題を解こうとする大多数のユーザにとって，数理計画の理論とアルゴリズムに精通することは難しく，条件が変更されたとき，アルゴリズムをいかに変更すべきかが難しい．しかも，仮に，アルゴリズムの変更ができても，それに応じてプログラムを修正するには大変な手間がかかる．

最近，知識工学研究の発展に伴って，計画業務の計算機システム化に再挑戦する動きが出てきている[4)]．知識工学は，スケジュール問題のように，数学的な定式化が困難な場合（正確な記述や厳密な解析が難しく，アルゴリズミックな解を持たない世界[5)]）に有効であると言われる．すなわち，定式化が困難な部分，および，計画条件変更に伴って変わる部分への対応ノウハウを知識として引き出し，ルール表現することによって，システムとしてのフレキシビリティを確保する．知識工学アプローチによるスケジューリング問題への取組みとしては，CMU の ISIS[6)]，阪大の FMS システム[7)]などが報告されている．ISIS はジョブショップスケジュールを対象としており，スケジュール問題をフレーム形式に表現している．また，後者はヒューリスティックなディスパッチングルールを IF THEN ルールで表現し，これによってシミュレーション的にスケジュールを実行する．しかし，計画問題をいわゆるフレームや IF THEN ルールだけで解こうとしても，実用規模の問題では良い結果を得られる保証はなく，また，組合せの爆発に対応できない場合も多い．

知識工学アプローチには，以上のように解決すべき課題が残されている．条件変更に対する柔軟性を確保し，しかも，計画の最適化も実現するために，知識工学手法と数理計画手法とを併用して，スケジューリング問題を解くことが有効であると，我々は考えている．知識工学と数理計画との組合せアプローチもいく

† Work Scheduling Method by Applying Knowledge Engineering Supported by Mathematical Programming Technique by KENZO KURIHARA (Systems Development Laboratory, Hitachi, Ltd.), KEIICHI HARA (Hitachi Microcomputer Engineering, Ltd.), TAKASHI KOBAYASHI (Systems Development Laboratory, Hitachi, Ltd.) and TATSUNORI SHIOMI (Systems Design Works, Hitachi, Ltd.).

†† (株)日立製作所システム開発研究所
††† 日立マイクロコンピュータエンジニアリング(株)
†††† (株)日立製作所情報システム工場

つか報告されている．Bruno らは，ジョブショップスケジューリングシステム[8]を報告している．このシステムは人間のノウハウをプロダクションシステムで表現し，これによる作業割付けを基本とし，割付けの良さの評価に待ち行列理論を活用している．また，Goldstein らによる NUDGE[9] はスケジュールの解法そのものは LP などの数理計画手法を用いているが，問題の定義部分に知識工学を応用している．ユーザを誘導し，解法に沿った定式化を完成させる．以上の例では，解法そのものは，知識工学手法または数理計画手法のどちらかを採用しており，計算速度あるいは解の実用性の面で問題が残されている．

本論文では，これら問題の解決を狙いとして，知識工学と数理計画の両手法を融合した，線形計画法の支援によるルールベース型作業スケジュール方式を提案する．スケジュール問題においては，定式化できない部分が残ることを根本的には解決できないため，本方式では，主導権をスケジューラ自身が持ち，スケジューリングの途中でいつでもスケジューラの意思を反映できる．第2章で対象とするスケジュール問題の特徴を述べ，第3章でスケジュール方式を提案する．さらに，第4章では純粋な知識工学方法と，これを改善した提案方式とを数値実験により比較評価する．

2. 作業スケジュール問題

2.1 作業スケジュール問題

作業スケジュール問題とは，複数の資源（人，機械など）に対して，複数の作業の中からどの作業を処理させるかを決定する問題である．このとき，処理能力，割付け規則などの多数の制約条件を満たす割付けを行う必要がある．例えば，各作業はその作業を処理する資格を持っている資源にしか割り付けられない．また，資格を行使しないと取り消されるような場合もあり，この場合には，資格消滅期限の近い資源に，資格維持のための作業を割り付けなければならない．さらに，資源が機械の場合には定期的な保守・点検を，人間の場合には研修などを，実作業の間をぬって割り付けなければならない．また，これらの制約条件は，それ自体がときどき変更されてしまうことが多い．

作業スケジュール問題は，与えられるデータ，制約条件，目的関数，決定変数を定義することによって定まる．例えば，**図1**に示すように定義できる（ここでは，日単位での1か月間のスケジュール立案を想定している）．すなわち，

①与えられるデータ：次の三種類である．

a．実施すべき作業（月間）

b．資源の月間予定（休日，等）

c．資源の割付け現況（何らかの理由により，当月のスケジュールの一部が既に定まっている分）

②制約条件：資源への作業の割付けに関する制約条件．絶対に守るべき条件（ハードな制約）と，守って欲しい条件（ソフトな制約）がある．

a．ハードな制約（保守・点検実施，資格確認，等）

b．ソフトな制約（各資源の総作業時間の上限，等）

③目的関数：資源間での作業時間の平準化，同種作業繰り返しの回避，などが挙げられる．

④決定変数：各資源の月間作業スケジュール

以上に示した作業スケジュール問題を形式的に定式化すれば，以下のように示される．ここでは，鉄道，航空機など運行スケジュールが決まっており（すなわち，作業日程が決まっている），運転手（資源）を割り付ける場合を考える．

初めに記号を次のように定義する．

i　：資源（$=1,2,\cdots,I$）

j　：作業（$=1,2,\cdots,J$）

$m^{(a)}$　：種類aに属する，保守・点検作業（$=1,2,\cdots,M^{(a)}$）

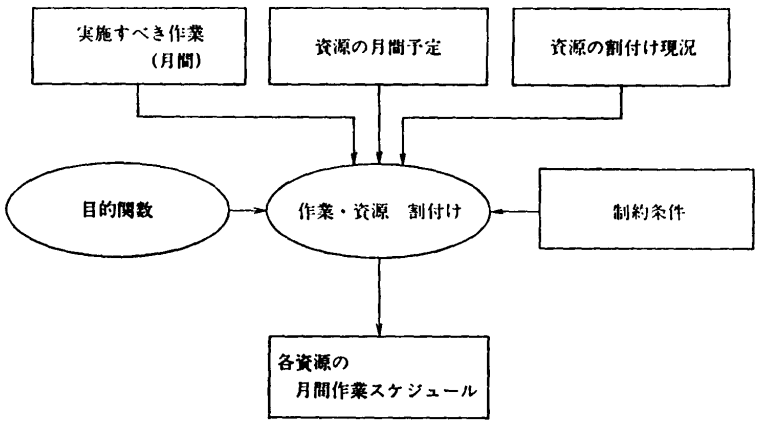


図1　作業スケジュール問題
Fig. 1　Work scheduling problem.

d_{ia} : 1…資源 i は種類 a の保守・点検作業が必要 / 0…資源 i は種類 a の保守・点検作業が不要

q_{ij} : 1…資源 i が作業 j の処理資格を持つ / 0…資源 i が作業 j の処理資格を持たない

s_{ij} : 1…初期段階では，作業 j の実行期間中，資源 i に予定が入っていない / 0…初期段階では，作業 j の実行期間中に資格 i に予定が入っている

$s'_{im^{(a)}}$: 1…初期段階では，保守・点検作業 $m^{(a)}$ の実行期間中，資源 i に予定が入っていない / 0…初期段階では，保守・点検作業 $m^{(a)}$ の実行期間中，資源 i には予定が入っている

$r_{jj'}$: 1…作業 j と j' を同一資源が処理可能（日程が重なっていない） / 0…作業 j と作業 j' を同一資源が処理不可（日程が重なっている）

$r'_{jm^{(a)}}$: 1…作業 j と保宇・点検作業 $m^{(a)}$ を同一資源が処理可能（日程が重なっていない） / 0…作業 j と保守・点検作業 $m^{(a)}$ を同一資源が処理不可（日程が重なっている）

$r''_{m^{(a)}m^{(a')}}$: 1…保守・点検作業 $m^{(a)}$ と $m^{(a')}$ を同一資源が処理可能（日程が重なっていない） / 0…保守・点検作業 $m^{(a)}$ と $m^{(a')}$ を同一資源が処理不可（日程が重なっている）

t_j : 作業 j の作業時間

$T_i^{\min}, T_i^{\max}$: 資源 i の総作業時間の下限値，上限値

$h_{m^{(a)}}$: 保守・点検作業 $m^{(a)}$ を割付け可能な資源の数（保守・点検作業の処理能力の上限）

$w^{(k)}$: 評価項目 k の重要度 $(k=1,2,\cdots,K)$

$c_{ij}^{(k)}$: 資源 i に作業 j を割り付けることを，評価項目 k の観点から評価したときの良さの程度

x_{ij} : 決定変数 1…資源 i に作業 j を割り付ける / 0…資源 i に作業 j を割り付けない

$y_{im^{(a)}}$: 決定変数 1…資源 i に保守・点検作業 $m^{(a)}$ を割り付ける / 0…資源 i に保守・点検作業 $m^{(a)}$ を割り付けない

このとき，作業割付け問題は例えば次式で定式化できる．

$$\sum_{i=1}^{I}\sum_{j=1}^{J}\left(x_{ij}\sum_{k=1}^{K}c_{ij}^{(k)}\cdot w^{(k)}\right) \rightarrow \text{Max.} \quad (1)$$

ここで，

$$\sum_{i=1}^{I}x_{ij}=1, \qquad j=1,2,\cdots,J \quad (2\text{a})$$

$$\sum_{m^{(a)}=1}^{M^{(a)}}y_{im^{(a)}}=d_{ia}, \quad \begin{matrix} i=1,2,\cdots,I \\ a=1,2,\cdots,A \end{matrix} \quad (2\text{b})$$

$$x_{ij}\leqq q_{ij}\cdot s_{ij}, \quad \begin{matrix} i=1,2,\cdots,I \\ j=1,2,\cdots,J \end{matrix} \quad (2\text{c})$$

$$y_{im^{(a)}}\leqq s'_{im^{(a)}}, \quad \begin{matrix} i=1,2,\cdots,I \\ m^{(a)}=1,2,\cdots,M^{(a)} \end{matrix} \quad (2\text{d})$$

$$x_{ij}\cdot x_{ij'}\leqq r_{jj'}, \quad \begin{matrix} i=1,2,\cdots,I \\ j=1,2,\cdots,J \\ j'=1,2,\cdots,J \end{matrix} \quad (2\text{e})$$

$$x_{ij}\cdot y_{im^{(a)}}\leqq r'_{jm^{(a)}}, \quad \begin{matrix} i=1,2,\cdots,I \\ j=1,2,\cdots,J \\ m^{(a)}=1,2,\cdots,M^{(a)} \end{matrix} \quad (2\text{f})$$

$$y_{im^{(a)}}\cdot y_{im^{(a')}}\leqq r''_{m^{(a)}m^{(a')}}, \quad \begin{matrix} i=1,2,\cdots,I \\ m^{(a)}=1,2,\cdots,M^{(a)} \\ m^{(a')}=1,2,\cdots,M^{(a')} \end{matrix} \quad (2\text{g})$$

$$T_i^{\min}\leqq\sum_{j=1}^{J}x_{ij}\cdot t_j\leqq T_i^{\max}, \qquad i=1,2,\cdots,I \quad (2\text{h})$$

$$\sum_{i=1}^{I}y_{im^{(a)}}\leqq h_{m^{(a)}}, \quad m^{(a)}=1,2,\cdots,M^{(a)} \quad (2\text{i})$$

上記の定式化は，典型的な例を示した．スケジュール結果の善し悪しを決定する評価項目は複数あり，それらは次元が異なる．提案方式では，各評価項目 k の観点から見たときの良さの程度 $c_{ij}^{(k)}$ をその重要度 $w^{(k)}$ で重み付けして，$\sum_{k=1}^{K}c_{ij}^{(k)}\cdot w^{(k)}$ によりスカラ値化した．ところで，上述の②b，③項に関するものは，スケジュールの途中結果を見て変えられることもある．すなわち，(1)式における $w^{(k)}$ は，スケジュール実行中にダイナミックに変化すると考えるべきである．また，s_{ij}, $s'_{im^{(a)}}$ および d_{ia} はスケジュール途中の状況によって1を0に変更すべき場合もある．同様に，$T_i^{\min}$, $T_i^{\max}$ も変更する場合がある．

以上に示した作業スケジュール問題は，線型計画法における割当て問題と類似している．しかし，次の点で線型計画法における割当て問題と異なる．すなわち，①資源に複数の作業を割り付けなければならない，②作業間に依存関係が存在する（資源にある作業を割り付けることの良否は，その資源にどの作業を既に割付け済かによって決まる），③作業割付けを邪魔しない範囲でできるだけ多くの資源に保守・点検，研修などを割り付ける．したがって，作業スケジュール問題は，単純に割当て問題として解くことができない．

2.2 作業スケジュール問題の特徴

スケジュール条件のすべてを事前に決定でき，スケジュール問題を明確に定式化できれば，数理計画手法で解ける．しかし，実際には，下記の理由により問題の定式化は困難である．

①スケジュール条件の事前決定が困難：スケジュール条件を事前にすべて決定できない．例えば，ある資源の総作業時間は，その上限値を越えてはならないが，作業日程，他の資源の保守・点検などの関係から，上限値を一部変更しなければならないこともある．

②問題自体が周囲の環境と共に変化：資源の増減，各資源の資格の変化，保有資源の能力などの環境条件によってスケジュール問題は変化する．したがって，割付け方法も変化に追従できなければならない．しかし，従来の方式では，割付けアルゴリズムがコーディングされているため，これを環境に合わせて変更することは実際上困難である．

このように，実際のスケジュール問題では，スケジュール条件の事前決定が困難であり，しかも，問題自体が周囲の環境により変化する．そのため，数理計画アプローチだけで解決することは困難である．また，スケジュール問題は各種条件を満たす順列組合せ問題であり，シミュレーションによって，単に試行錯誤を繰り返しても適切な解を得られる保証はない．

3. 作業スケジュール方式

3.1 基本的な考え方

3.1.1 作業スケジュールシステムの狙い

作業スケジュールシステムの狙いは，次の三点に集約できる．

（1） システムの成長性

スケジューリング上の制約条件は，ときどき変更される．また，資源や作業に関する特性の変化に伴って，作業割付けノウハウは変化する．システムは，これらの変更，変化に容易に対処する．

（2）『質』の良い割付け計算を高速に実現

作業スケジュール問題は，膨大な順列組合せの中から，多数の制約条件を満足する組合せを求める問題である．最適割付け計算を実用的な時間で行うために，探索とバックトラックの繰り返しなどの単純な方法は避ける．資源間でバランスのとれた作業割付けを高速に計算する方法を開発する．

（3） スケジューラの創造的業務の支援

解が求まらないときには，割付け状況に応じて制約条件を一時的に緩和する．スケジューラ自身による，このような創造的判断の業務を始めとして，スケジューラが主導権を持った Computer Aided Planning を指向する．自動スケジュールは，システムの重要な機能の一つであるが，これがすべてではない．

3.1.2 作業スケジュール方式の考え方

スケジュール問題を整数計画法や分枝限定法などの数理計画アプローチだけで解決することは，前述のように困難である．数理計画アプローチでは，アルゴリズム開発が強調されすぎており，問題領域固有の知識の重要性の認識とその利用が不十分である[10]という反省もある．一つの解決方向として，近年，人工知能の研究成果を実用化する提案が，知識工学の名の下に進められている．このアプローチは，検討対象となる問題の定式化が困難な場合に有効である．作業スケジュール問題も，この種の問題の範疇であり，本論文では，プロダクションシステムに基づく知識工学アプローチによって問題の解決を図る．

（1） 狭義のエキスパートシステムによる解決の限界

プロダクションシステムでは，エキスパートが有する対象に関する知識を“IF（条件）THEN（結論）”形式で記述し，知識ベース化する．そして，三段論法型の推論機構によって，ある条件が与えられたときの結論を導く．この方法の利点は，①構成が単純，②計画システムの保守性が従来言語によるシステムより向上（知識と推論とが分離），③プロトタイプ開発の迅速化（エキスパートシステム構築ツールの活用），などにある．しかし，欠点として，①処理時間が遅い，②対象データの特性が変化した時に解の善し悪しが発生する，など性能上の問題が挙げられる．そのため，スケジュール問題のような組合せ問題に対して，狭義のエキスパートシステムによるアプローチだけででき

る範囲は狭い.

上記の種々の問題に対する一つの解決法は，次に示すように，エキスパートの知識と解法アルゴリズムを融合して活用することである.

(2) 知識とアルゴリズムを融合したシステム

スケジューラが実際に割付けを実施する際，基本的には時間の流れに沿って順次割り付けていく．ただし，常に先を見て，特殊な状況にある資源や作業があれば，それに着目して先に割り付ける．すなわち，『現状の割付け状況から部分問題を抽出し，それを解く．その結果に基づいて次の部分問題を探す』というサイクルを繰り返す．ここで，部分問題の抽出は，対象に固有のノウハウである．そこで，このノウハウの活用をプロダクションシステムで実現する．また，部分問題は線形計画法における割当て問題などとして定式化する．これにより，複雑なスケジュール問題を汎用的で簡略な割当て問題の組合せで解くことができる．そして，スケジュール立案ノウハウや制約条件が変化しても，これに対処する保守性を確保すると共に，割付け計算の最適化と高速化を実現できる．また，このように対象に固有のロジックを数学的な計算ロジックから分離しておくこと，および，処理時間を高速化することは，計画立案者が主導権を持った Computer Aided Planning を実現する上で重要なことである.

割付け方式の基本的な考え方を図2に示す．提案手法では，計画立案者のノウハウを知識ベースに格納し，この知識ベースに基づいて，自動的にスケジューリングを実行する．しかし，満足のいくスケジュール結果を全自動で常に得ることは望めない．そこで，図に示したように，知識工学アプローチ部は，スケジューラ自身がとってかわってスケジューリングを行うことができる．すなわち，部分問題を抽出し，その解を求める度に計画立案者が介入し，その時点までのスケジュール結果を評価することができる．この評価の結果，割付け結果の取消し，あるいは，割付け優先範囲とその解法の指定を任意に指定できる.

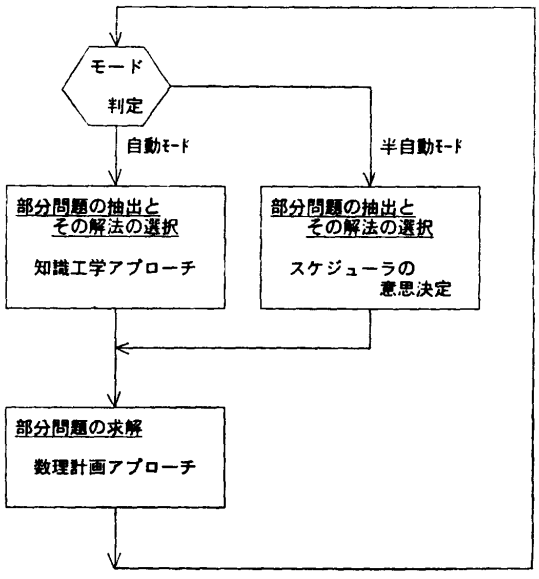


図 2 作業割付け方式の考え方
Fig. 2 Idea of work scheduling method.

3.2 スケジュール方式

スケジューラが介在する割付け方法を半自動モードと呼ぶ．この半自動モードは，スケジューラが正に自由にスケジュールすることができる．一方，知識ベースを用いて，知識工学アプローチ部をシステムが実行する方法を自動モードと呼ぶ.

3.2.1 割付け手順

自動モードでは，一つの資源にたかだか一つの作業しか割り付けられないように，割付け期間を『サブ期間』に分割する．すなわち，サブ期間に開始される作業のうち，最も早く終わる作業の終了日とサブ期間の最終日を一致させる．そして，このサブ期間ごとに，線形計画法における割当て問題の解法アルゴリズムにより作業割付けを実行する．しかし，サブ期間に分割して問題を解くため，作業間の依存関係を完全に考慮することはできない．そのため，本来は解が存在するのにその解を見つけられない危険がある．これに対処するために，知識工学手法を次のように活用する.

あるサブ期間の作業割付けを実施後，先の期間を見て，割付けが難しいと思われる部分を部分問題として抽出し解く．この部分問題の抽出とその解法の選択は，スケジュール専門家のノウハウであり，これらを知識工学手法におけるプロダクションシステムを利用したルールベースシステムで実現する．このように，難部分問題の抽出・求解とサブ期間の割付けを繰り返すことによって，全資源の割付け対象期間（例えば，月間）の作業スケジュールを作成する．この割付け手順の概要を図3に示す．図に示すように，(1)割付け状況の認識，(2)戦略立案，(3)割付け実行，のサイクルを繰り返し，最適割付けを進めていく．半自動モードでは，(1)(2)をスケジューラ自身が判断するが，自動モードでは次のようにして実行する.

どの部分が難しい部分問題であるかは，専門家が経験的に体得している．この経験的知識を戦略ルールと

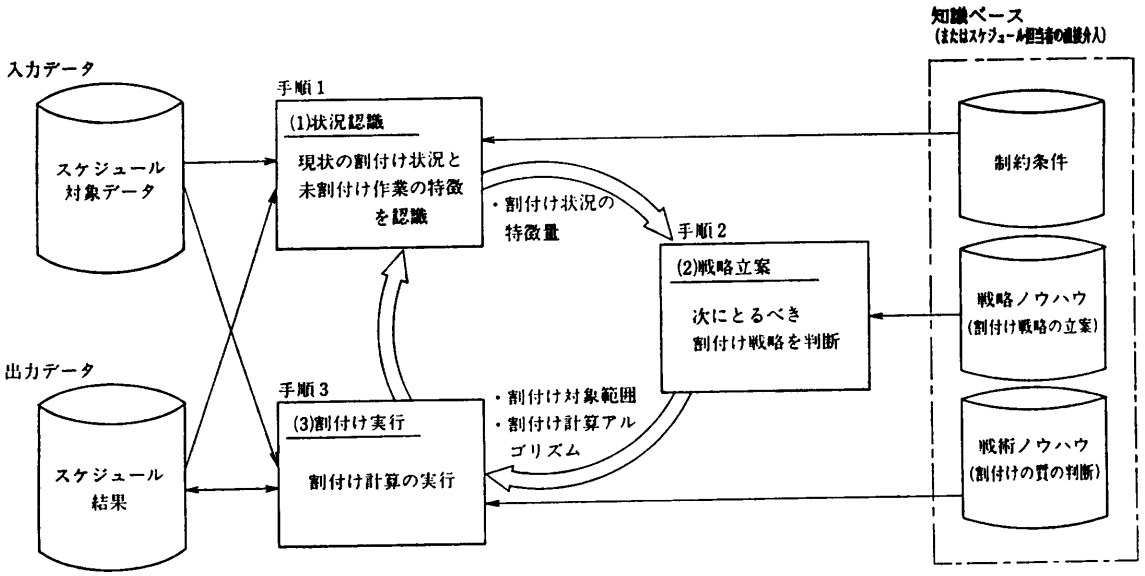


図 3　作業割付け手順
Fig. 3　Procedure of work scheduling.

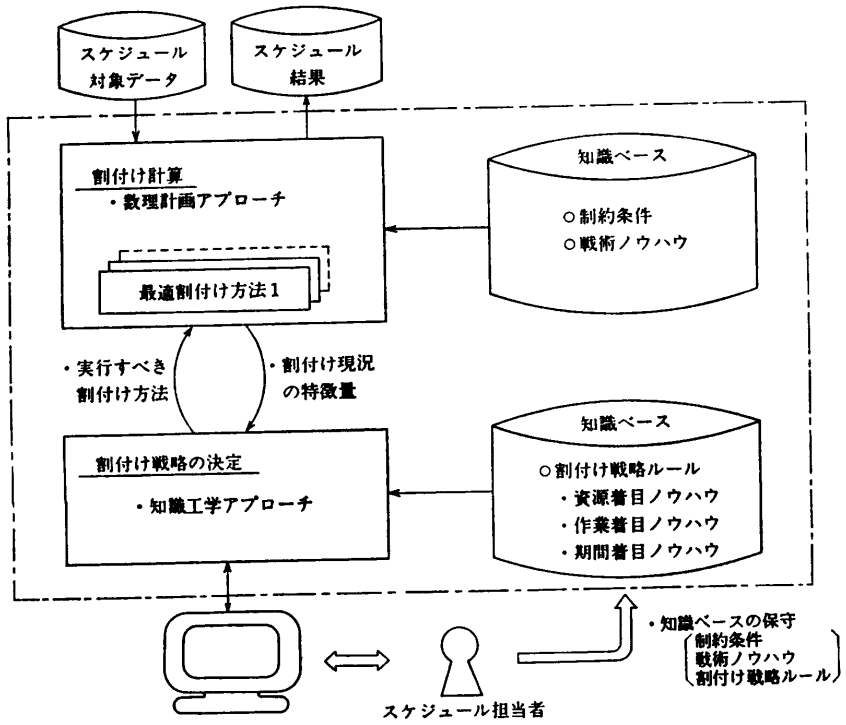


図 4　提案方式によるシステム構成の概念図
Fig. 4　System architecture outline based on the proposed method.

して表現し，自動的に難部分問題の抽出を行う．この難しい部分の抽出とその解法の選択（割付け戦略）を知識工学手法で実施する．そして，『難しい部分およびサブ期間の割付け』といった部分問題の解は，割当て問題に対する解法アルゴリズムなどで求解する．本方式によるシステム構成の概念を図 4 に示す．

具体的な割付け戦略としては，次のものを考える．ただし，これらはノウハウ的な知識であり，随時，変更できるようにしておく必要がある．

（a）　特殊な状況にある資源，作業を対象にした割付け

①　作業着目の戦略

a．候補資源数による部分問題の抽出：候補資源数の少ない作業から，順に割り付ける．

b．作業所要日数による部分問題の抽出：所要日数の長い作業から，順に割り付ける．

②　資源着目の戦略

a．各種資格維持用の候補作業数による部分問題の抽出：今月中に作業処理資格が切れる資源のうち，その資格を維持するための候補作業数が少ない資源から，順に割り付ける．

b．各種資格維持期限による部分問題の抽出：資格の期限切れが近い資源から，順に割り付ける．

c．処理可能作業数による部分問題の抽出：処理可能な作業数の少ない資源から，順に割り付ける．

(b) 特殊な状況にない場合の割付け

③ 期間着目の戦略

上記のいずれの戦略もとれない（各戦略とも，資源数，作業数の上下限値を設け，その範囲内の部分を抽出）場合，前述のサブ期間ごとに割り付ける．この際，着目するサブ期間に実施すべき全作業を一括して最適に割り付ける（期間着目割付け）．

3.2.2 割付けノウハウの分類とその表現法

スケジューラの持つ割付けノウハウをいかにシステムに組み込むかが，一つの大きな課題である．割付けの三ステップ（図3）に対応して，割付けノウハウも三分類した．そして，各ノウハウに適した表現法を設定した．これによって，外部環境の変化に対する，システムの成長性を確保した．

割付け状況の認識には，制約条件を利用する．制約条件はフレームで表現する．具体的には，各制約条件を日本語表現し，変更される部分はパラメータとして容易に変更できるようにする．ここで，制約条件は，条件項目そのものが増加する際は，条件チェックルーチンを作り，組み込む必要がある．しかし，一般には，項目そのものの増加は少なく，条件のパラメータ値が変わることが多く，これについてはその値を変更するだけでよい．

戦略ノウハウは，割付け状況の認識に基づき割付け戦略を立案するためのノウハウである．ここで，割付け戦略とは，どの範囲の資源あるいは作業に着目して，どのアルゴリズムを実行するかを決定することである．この戦略立案ノウハウは，IF THEN ルール形式で表現する．IF 側には割付け状況の特徴量を，THEN 側にはその時にとるべき戦略を示す（具体例は4.1節に示す）．この IF THEN ルールはルール間の相互関係が少ないため，修正は容易である．

戦術ノウハウは，割付けの質を判断するための知識である．一般に，質を判断するための項目は変化することが少なく，各項目をどれだけ重視するかが，環境条件によって変化する．そこで，各項目の重要度をウェイト（前述の $w^{(k)}$）で数値表現する．このウェイトを変更することによって，割付け実行時に重視したい事項を変える．

なお，上記の各知識ベースに対して，それぞれ専用のエディタを用意してあり，入力操作そのものの容易化を図った．

3.2.3 期間着目アルゴリズム

抽出した部分問題の最適解を求めても，それが全体としての最適解であるとは限らない．したがって，部分問題の境界部分の条件を十分に考慮して解を求めなければならない．考慮しないと，次のステップの部分問題を解く際に既割付け部のやり直しが多発し，各部分問題の最適解を求めることが無意味となる．

割付けアルゴリズムの基本となる，期間着目アルゴリズムでは，上記の課題を次により解決している．すなわち，下記(1)のようにして，割付け期間の最適化を図ると共に，下記(2)のようにして，割付けのやり直しの発生を防止する．

(1) 割付け期間の最適化の考え方

各作業に対して，その作業を処理可能な資源をすべて抽出する．そして，各資源にその作業を割り付けることが，どの程度よいことであるかを総合的に評価する．この評価には，前述の戦術ノウハウを利用し，スケジューラが実際に評価に使っている項目とその重要度を反映させる（**図5(a)**）．そして，評価点の合計が最大となる組合せを探索し，その組合せを最適割付けと考える．この最適計算は，具体的には数理計画における．割当て問題[11]として次のように定式化し，求解する．次式においては，$I \neq J$ の場合に限って，ダミーの資源 i または作業 j を導入し，資源数と作業数を等しくする．すなわち，$N \triangleq \max.(I, J)$ とする．なお，ダミーに対応する $c_{ij}^{(k)}$ はすべて同じ値にする．

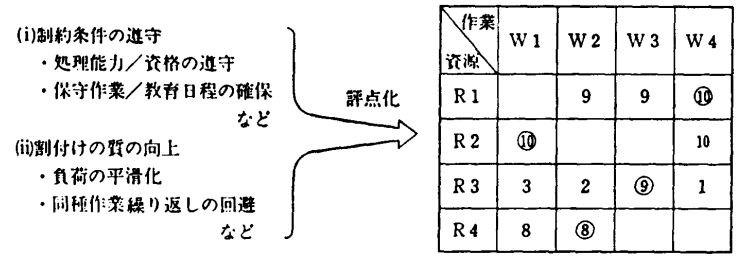
$$\sum_{i=1}^{I}\sum_{j=1}^{J}\left(x_{ij}\sum_{k=1}^{K}c_{ij}^{(k)}\cdot w^{(k)}\right) \rightarrow \text{Max.} \quad (3)$$

ここで，

$$\sum_{j=1}^{N} x_{ij}=1, \quad i=1,2,\cdots,N \quad (4\,a)$$

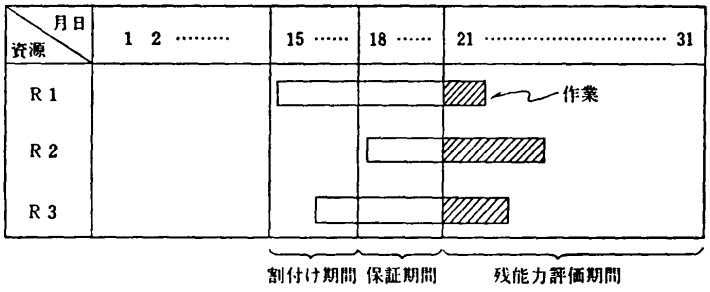
$$\sum_{i=1}^{N} x_{ij}=1, \quad j=1,2,\cdots,N \quad (4\,b)$$

上記の定式化では，(2b)～(2i)式の制約条件も評価項目 k に含め（ただし，日程の重なり回避に関する条件(2e)(2f)(2g)式については，期間を分割したことによって考慮するので除く），単純な割当て問題でモデル化した．重要度 $w^{(k)}$ は，本来，異なる次元の評価項目をスカラ化するためのウェイトである．したがって，定性的には，重視したい項目のウェイトを



注）○印は，最適割付け結果

（a）割付け期間の最適割付け
（複数作業の割付けを総合的に判断）



注1）保証期間：近い将来への影響を考慮
・保証期間内の全作業に対して，処理可能資源を確保
注2）残能力評価期間：遠い将来への影響を考慮
・残能力評価期間にかかる作業の足の部分で処理できなくする作業数を最少化する。

（b）最適割付けの破壊防止
（割付けのやり直しの発生防止）

図5　期間着目アルゴリズム
Fig. 5　Scheduling algorithm.

大きくすればよいが，具体的にいくつにすべきかを理論的に決めることはできない．作業者の人数，保有資格，処理すべき作業の量などによって，スケジュール問題の難しさが変化するが，これに伴って，ウェイト $w^{(k)}$ も変わる．このように重要度 $w^{(k)}$ は対象とする問題ごとに決めるべきものであり，変更する際には，計画立案者がスケジュール結果を見て，試行錯誤によって値を決める．

（2）　最適割付けの破壊防止

割付け期間に隣接する近い将来を保証期間と名付け，この保証期間に実行すべき全作業に対して，資源を確保しておく．割付け期間に開始される作業は，そのほとんどが保証期間に足を伸ばしているため，上記のように保証期間の割付け資源を確保しておかないと，割付けのやり直しばかり発生してしまう（図5（b））．

さらに先の期間は，残能力評価期間と名付け，先を予測評価する．すなわち，この期間にかかる足の部分で，本来，処理できるはずの作業をいくつ処理できなくなるかを計算し，これをできる限り少なくすることによって，将来の処理能力の無駄使いを削減する．具体的には，処理できなくなる作業数を，（3）式における評価項目 k の一項目として評価する．

なお，作業着目，資源着目アルゴリズムも，期間着目アルゴリズムと基本的には同じである．それぞれ，割付け対象とする資源あるいは作業の範囲が異なる．

3.2.4　割付けのやり直し方法

割付けできない作業が発生した場合，一部の割付けをやり直す．システムが自動的に実施する場合と，スケジューラ（担当者）が介在して実施する場合とがある．

システムによる自動やり直しは，直前の期間着目割付けをすべて取り消し，保証期間と割付け期間を少しずつ短くして，割付け不能が解消されるまで試行錯誤する．もし，解消されなければ，難問題として事前に割り付けられた部分を取り消す．これにより，その後のスケジュールの進行によって，別の割付けをしたほうがよい状況におちいっている場合に対応できる．なお，条件違反でも状況によっては，割り付けざるを得ない場合もある．そこで，自動モード時は，違反内容と割付け候補を示し，スケジューラに割付けの可否を問い合わせる．やり直し方法にも種々のノウハウがあり，これを IF THEN ルールで表現し，知識ベース化することは可能と思われるが，これは今後の課題である．本論文では，ここはスケジューラに判断させ，実行するためのマンマシンインタフェースを提供する．

自動やり直しで割付け不能が解消しない場合，スケジューラが介在して調整する．この場合には，スケジューラは任意の作業，期間についてやり直す．その後，再度，自動モード割付けの続行も可能である．

4. 提案方式の特性に関する検討

提案方式に基づき，試作システムを16ビットワークステーション（日立 2050）上に開発し，提案方式を評価した．なお，試作システムは，C言語とエキスパートシステム構築ツール ES/KERNEL を利用して開発した．C言語部は約 7 kstep, ES/KERNEL 部は約 20 ルールである．開発には，約 15 人月を要した．

4.1 検討対象モデル

作業者（資源）への1か月間の作業割付けを検討する．各作業は数日間に渡って実施され，誰が実行しても作業時間は同じである．ただし，各作業を実行する作業者には，それぞれの作業に対応していくつかの資格が必要である．また，スケジュール結果の善し悪しは，①割付け不可となる作業の低減，②作業時間の平均化，③同種作業の繰り返しの回避，で評価する．このモデルは，例えば，鉄道，航空機，トラック等，運行スケジュールが決まっていて，どの運転手を割り付けるかを計画する場合を一般化したものである．なお，簡単のため，すべての i，a に対して，$d_{ia}=0$ とした．

本章で対象とする問題モデルは，提案方式の特性を検討するために設定したものであり，基本構造だけからなる簡略化したモデルとした．したがって，割付け戦略も簡略化することができ，次の三戦略を用意した．

戦略 1：作業着目の戦略
IF 処理可能作業者数≦[M_1] の作業がある
THEN 対象＝[当該作業群] として
アルゴリズム＝[作業着目] を実行する

戦略 2：作業者着目の戦略
IF 処理可能作業数≦[N_1] の作業者がいる
THEN 対象＝[当該作業者達] として
アルゴリズム＝[作業者着目] を実行する

戦略 3：期間着目の戦略
IF 処理可能作業者数≦[M_1] の作業がない
and 処理可能作業数≦[N_1] の作業者がいない
THEN 対象割付け/保証期間＝[D_1/D_2] として
アルゴリズム＝[期間着目] を実行する

これらの戦略は，ノウハウとして得られるものである．しかし，本検討モデルに対して，そのまま対応するノウハウはない．そこで，本検討モデルのベースとした具体問題で集めたノウハウに基づき，上記の戦略を設定し，$M_1=1$，$N_1=3$ とした．なお，D_1，D_2 は，下記のように種々の値について実験した．

4.2 スケジュールパラメータとスケジュール結果の質の関係

本節では，スケジュール結果の質に対する，割付け戦略の効果を検討する．さらに，スケジュールパラメータのなかで最も重要である，割付け期間および保証期間と，スケジュール結果の質との関係を検討する．

4.2.1 割付け戦略の効果

期間着目アルゴリズムを実行するときの割付け期間として，その期間内に一人の作業者にはたかだか一つの作業しか割り付けられない期間（前述のサブ期間）を設定した．本評価実験に使用したデータ（作業者 70 人，作業数 230）では，平均 3.1 日であった．保証期

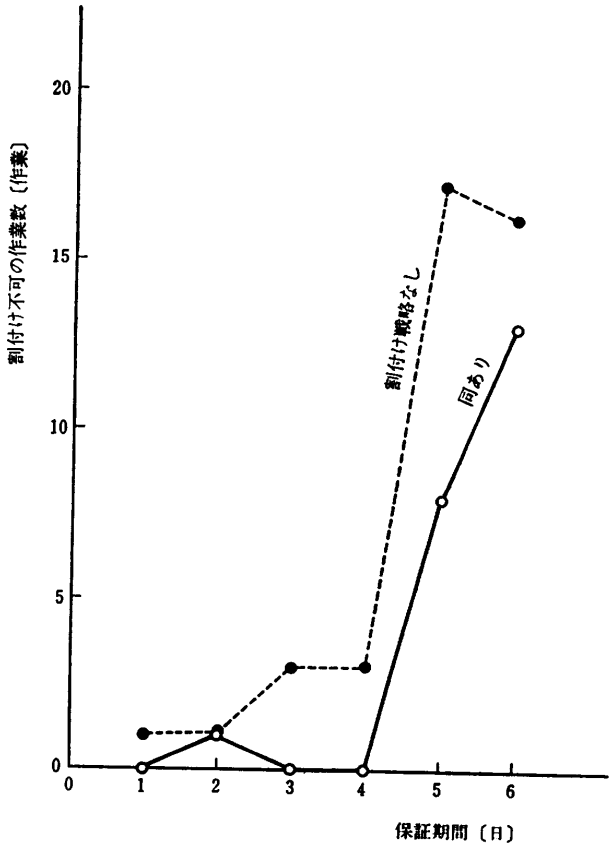


図 6 割付け戦略の効果
Fig. 6 Effect of assigning strategy.

間を種々に設定し，前記の三戦略を使わない場合（期間着目アルゴリズムだけで割付けを実施）と戦略を使った場合とで，割付けできなかった作業の数を調べた．その結果を，図6に示す．図に示すように，戦略を使うことによって，割付けできない作業の数を低減できる．しかも，戦略を使うことによって，保証期間の広い範囲に渡ってよい解が得られる．

4.2.2 割付け期間・保証期間とスケジュール結果の質との関係

割付け期間と保証期間を種々に設定して実験した．保証期間の日数をパラメータとして，割付け期間の日数と割付け不可の作業数との関係を図7に示す．ここでは，前記の三戦略を使った．割付け期間の日数の広い範囲に渡って安定してよい解が得られている．以上の結果（図6，図7）から，割付け期間，保証期間とも，広い範囲に渡って安定してよい解が得られることが分かる．今回の対象データでは，割付け期間は5日以内，保証期間は2日とするのがよい．

一般にこれらの期間をどの程度にすべきかを検討する．1か月分のスケジューリング実験では，期間着目アルゴリズムは複数回実行されるが，各回ごとに『負荷（割付け期間＋保証期間に含まれる作業数をその期間に作業実施可能な作業者数で割った値）』を求め，各実験ごとに平均値を求めた．この平均負荷と割付け不可となった作業数の関係を図8に示す．図に示すように，平均負荷が1を越えると急激に割付け不可の作業数が増加する．一方，0.9以下の場合にも，割付け不可の作業数は増加する．これは次の理由によると思われる．割付け期間と保証期間の

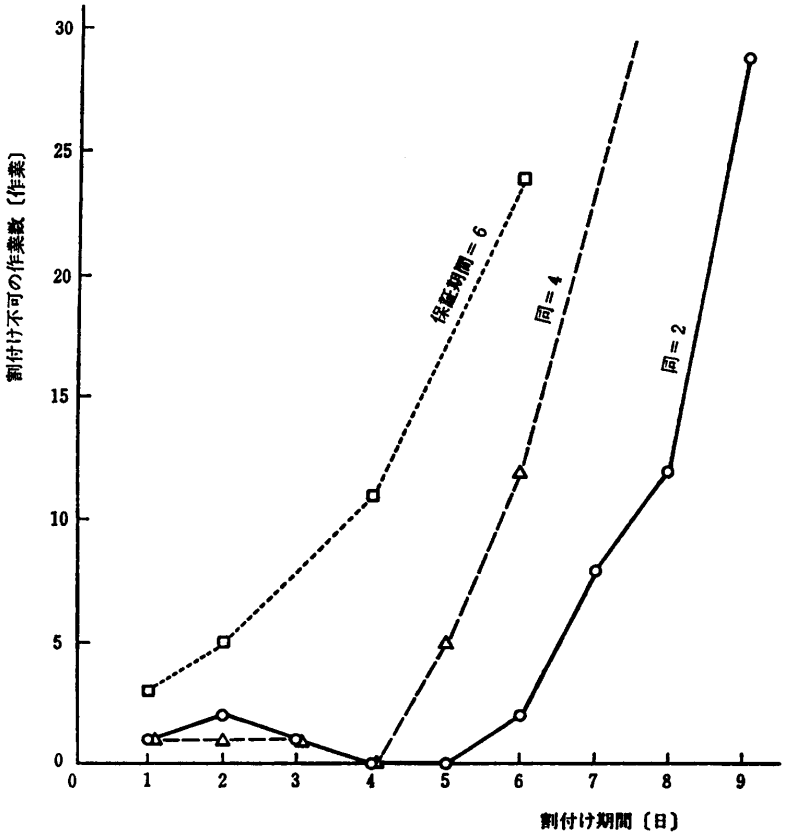


図7　割付け期間/保証期間に対する割付け不可作業数の変化
Fig. 7　Un-assigned work number vs. change of individual length of assigning and insurance term.

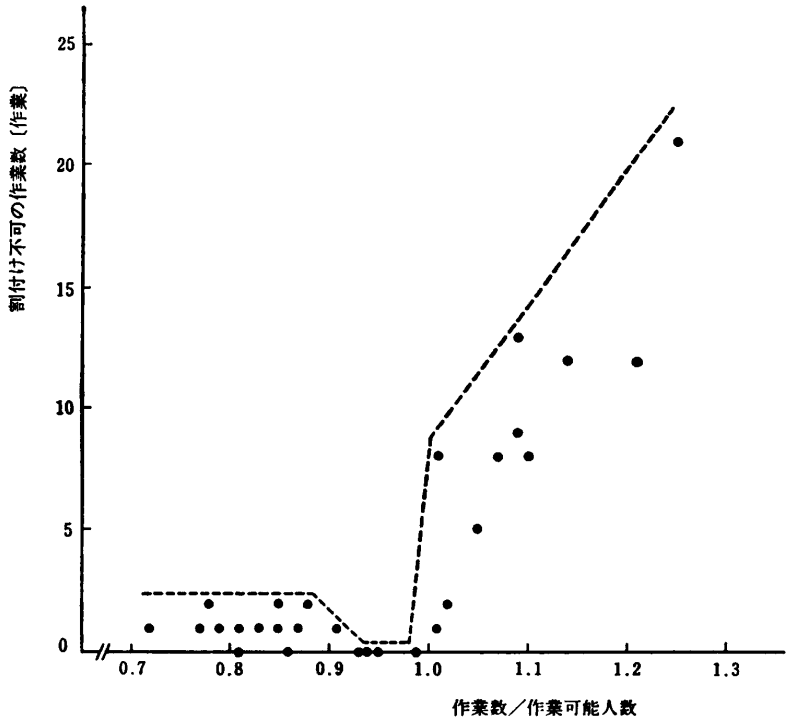


図8　平均負荷に対する割付け不可作業数の変化
Fig. 8　Un-assigned work number vs. average load.

合計日数が短いと，その間に開始される作業は少なくなるが，その期間に日程が空いている作業者の数は合計期間の長短によってあまり変化しない．そのため，合計期間が短くなると共に平均負荷は小さくなり，ほぼ比例関係にある．したがって，平均負荷が小さくなるのは考慮する期間が短くなる場合であり，この場合には先の期間の作業に対する資源確保ができないため，割付け不可の作業が発生しやすくなる．

以上の検討により，作業自体のタイムスケジュールが事前に決められている作業を対象として，一人の作業者に多数の作業を割り付けなければならない問題を提案方式で解く場合，平均負荷が0.9から1.0となる程度の期間に分割することがよい．すなわち，これが割付け期間と保証期間を決めるための指標になっていると考えられる．

4.3 知識工学単独方式との比較評価

提案方式は，知識工学アプローチの改善を狙っている．そこで，知識工学手法だけによる典型的な割付け方式と比較することによって，提案方式の特性を評価する．評価項目は，(1)計算時間，(2)スケジュール結果の質，(3)システムとしての保守性，の三項目とした．結果の概要を**表1**に示し，以下，各項目ごとに検討する．なお，知識工学手法だけによる方式では，スケジュールノウハウをルール表現した．本検討では，具体的には，以下に示すノウハウを用いた．

＜スケジュールノウハウ＞

①日付順に割り付ける．

②同一日の作業は処理可能作業者数の少ないものから割り付ける．

③各作業の処理作業者は次の優先度により割り付ける．

優先度1：その作業を割り付けることによって同種作業の繰り返しにならない作業者．

優先度2：総作業時間が少ない作業者．

④割付け不可の作業が発生したらバックトラックする．すなわち，その割付け不可作業を処理する資格を持っている作業者のうち，最も処理可能作業数が多い人の既割付け作業を取り消す．

4.3.1 計算時間

知識工学単独方式は，一作業ずつ割付けを実行する．このとき，ルール表現されたノウハウにより，適切な割付けを行っていく．しかし，この方法ではどうしても割付けのやり直し（バックトラック）をせざるを得ない状況が頻繁に起こる．そのため，計算時間が長く，しかも，バックトラックの発生状況によって計算時間は変動する．一方，提案方式では，数理計画の割当て問題アルゴリズムを利用して，複数作業を同時に最適に割り付ける．そのため，割付けのやり直しを

表1 提案方式の特性
Table 1 Evaluation result of proposed method.

方式 / 評価項目	KE・OR 融合方式（提案方式）	KE 単独方式	OR 単独方式
方式	＜割付け方式＞ ●柔軟性の確保 ……KE アプローチ ●最適割付けの高速計算（複数作業を一括割付け） ……数理計画アプローチ ＜言 語＞ ●KE アプローチ部 ……ES/KERNEL など ●数理計画部 ……C言語など	＜割付け方式＞ ●1作業ずつの逐次割付け方式 ＜言 語＞ ●ES/KERNEL, LISP など	＜割付け方式＞ ●月間を数個のサブ期間に分割 ●サブ期間ごとの一括割付け方式 ＜言 語＞ ●C言語など
1. 計算時間（高速性，安定性）	○ 安定・高速	× 不安定	○ 安定・高速
2. スケジュール結果の質	○ 制約条件の厳しさによらず質のよいスケジュールを実現	△ 1作業ずつの割付けのため，制約条件の厳しさにより，解の善し悪しが発生	× 割付けノウハウの記述が困難なため，スケジューラによる最終調整が多発
3. システムの保守性（割付けノウハウの変更）	○ 環境状況の変化に伴う割付けノウハウ（戦略ルール）の変更が容易	△ ルールと割付け計算とを分離することが容易でない	× プログラムのコーディングの変更 問題の再モデル化
総合評価	○	△	×

ほとんど発生させることなく，高速にしかも問題規模に応じた一定の時間でスケジュールを実行できる．

数値実験の結果を**図 9** に示す．横軸は問題の規模をスケジュール対象作業者の人数で表す．図に示すように，提案方式では，問題規模が拡大しても高速かつ安定した時間でスケジュールができる．図では省略したが，約 200 人に約 700 作業を割り付ける場合には，約 30 分を要した．これに対して，知識工学単独方式では，規模の拡大に伴ってスケジュール計算時間は指数的に長くなり，ばらつきも拡大した．図から分かるように，200 人による大勢のスケジュールは実用的な時間で実行できない．実線は提案方式による平均値を，点線は知識工学単独方式による平均値を示す．

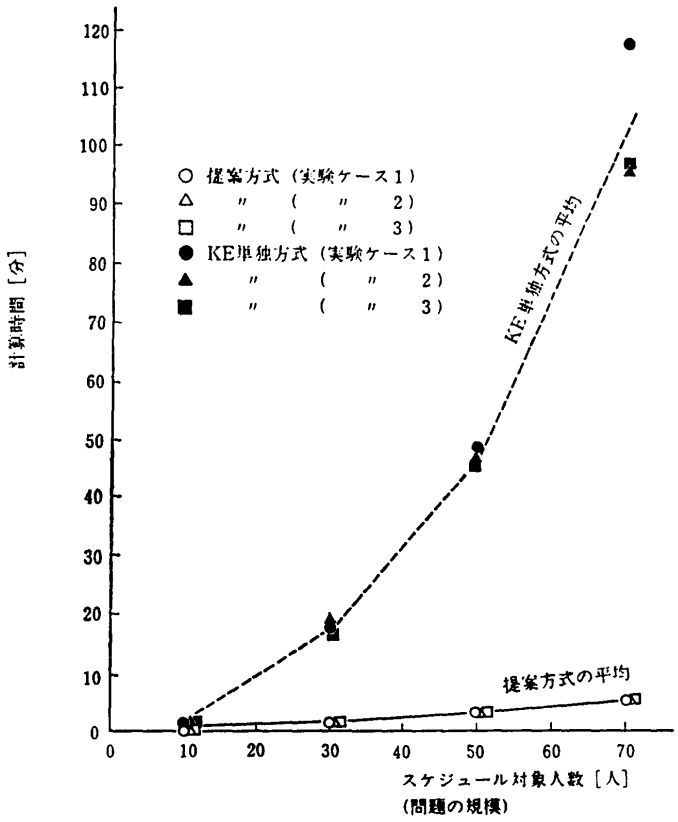


図 9 計算時間
Fig. 9 Calculation time.

4.3.2 スケジュール結果の質

知識工学単独方式では，一作業ずつの割付けのため，対象データによってはよい解が求まらないこともある．一方，提案方式では，スケジュール条件の厳しさによらず，質のよい解が期待できる．数値実験による検討結果を以下に記す．なお，スケジュール結果の質は，4.1 節に示したように次の三項目で評価する．

（1） 割付け不可作業の低減

対象問題の規模を 4 通りに変えて実験した（**図 10**）．知識工学単独方式では，問題規模の増大に伴って解の探索範囲が相対的に狭くなるため，割付け不可の作業が増加した．これに対し，提案方式では，問題規模にかかわらず，割付け不可の作業を低減できた．

（2） 作業時間の平均化

本検討モデルでは，作業の所要日数とは別に実質の作業時間を設定し，この作業時間の合計値を各作業者間で平均化した．各作業者の総作業時間の標準偏差を**図 11** に示す．知識工学単独方式，提案方式の間に，顕著な差はない．しかし，両方式の間では，前述のように割付け不可作業の数が異なっており，知識工学単独方式では，作業の割付けを控えることによって作業時間を平均化していると考えるべきである．

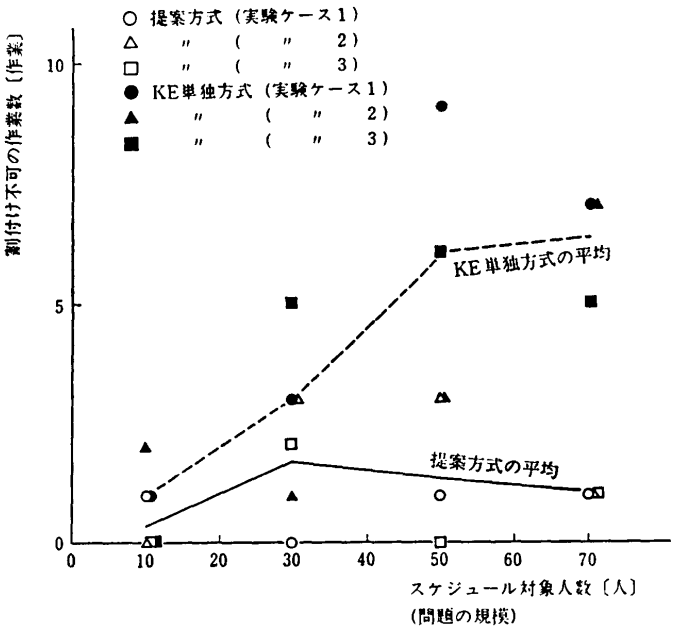


図 10 割付け不可の作業数
Fig. 10 Un-assigned work number.

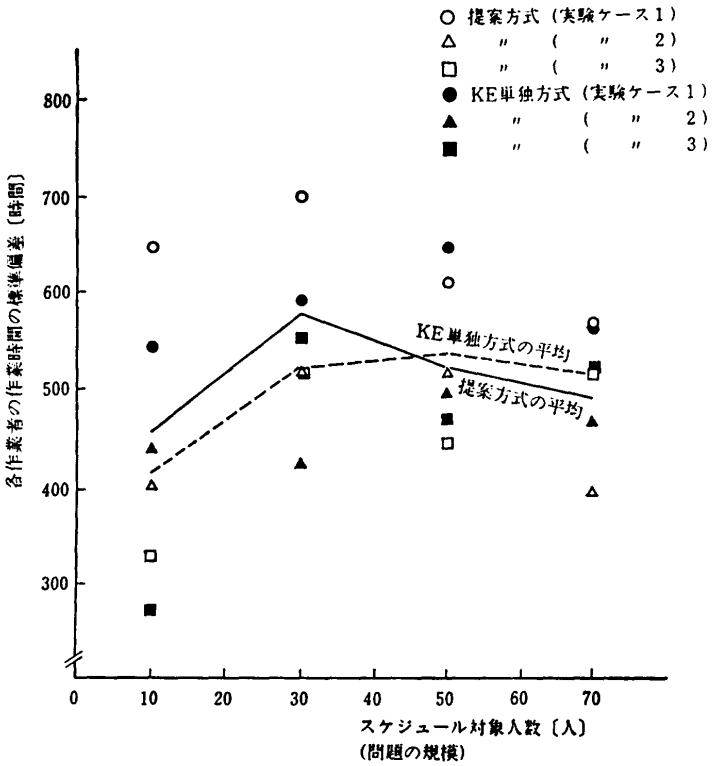


図 11 各作業者の作業時間の標準偏差
Fig. 11 Standard deviation of each worker's work time.

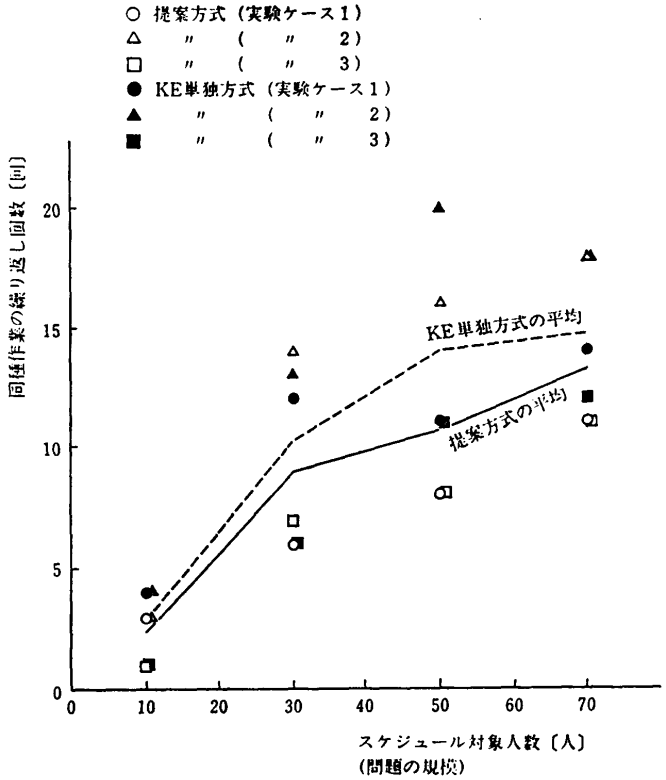


図 12 同種作業の繰り返し回数
Fig. 12 Repetition number of same kind work.

（3） 同種作業の繰り返しの回避

同種作業を同じ作業者に連続して割り付けないようにした．両方式による結果を図12に示す．同種作業の繰り返しは，当然，問題規模に比例して増加する．知識工学単独方式の方が悪い結果となったが，顕著な差はない．しかし，（2）と同様に知識工学単独方式では，作業の割付けを控えて繰り返しを回避している．

なお，提案方式により作業者が200人のスケジュールを実行した場合，割付け不可の作業者は零，総作業時間の標準偏差は570分，同種作業の繰り返し回数は37回であった．このことから，スケジュール対象規模が，大きくなってもスケジュール結果の質は良好であることを確認した．

4.3.3 システムの保守性

提案方式では，①割付けの論理（最適化アルゴリズム）と，②割付け問題の分割ノウハウ（割り付ける範囲と割付けアルゴリズムの選択）を分け，プログラム構造上も分離させた．これによって，それぞれの管理・保守部門も分けることができ，前者はシステム部門の人が管理すべきであり，後者はスケジュール担当者が自分で管理すべきである．対象データの変化に伴って変えるべきものは，一般に後者の方であり，最適化アルゴリズムそのものは変わらない．今回の実験の場合，割付け問題の分割ノウハウはルール表現しているが，処理の流れを制御するためのものも含めて約20のルールであり，保守が容易である．

一方，知識工学単独方式によるプログラムは，上記①②の両者をルールで表現した．簡略化した知識を用いたにもかかわらず，そのルール数は約80にもなり，ルール間の整合性をとることは容易でない．

5. おわりに

スケジュール条件の変更に対する柔軟性と最適スケジュール作成の高速処理を狙いとして，知識工学手法と数理計画手法とを融合した作業割付け方式を提案した．本方式では，割付け計算用に数種の最適計算ア

ルゴリズムを用意すると共に，割付け戦略を設定するためのノウハウを知識ベース化した．そして，割付けを実施する過程で，知識ベースにより，部分問題の抽出とこれを解くのに適したアルゴリズムの選択を行う．

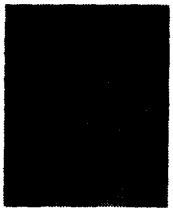
鉄道，航空機など，運行スケジュールが決まっていて，運転手の割付けを計画する場合を一般化したモデルを用いて，提案方式の特性を評価し，狙いどおりの効果があることを確認した．

謝辞 スケジュール問題の課題に関して御指導，御討論を頂いた(株)日立製作所情報システム工場橋田康明部長，五十嵐和夫主任技師に深い謝意を表します．また，本研究の機会を与えて頂くと共に，終始，暖かい御指導と御討論を頂いた当社システム開発研究所堂免信義所長，春名公一副所長，明石吉三博士，鷹田憲久博士に感謝いたします．

参考文献

1) R. W. コンウェイ(関根智明訳)：スケジューリングの理論，日刊工業新聞社 (1971).
2) 特集：スケジューリング，オペレーションズリサーチ，Vol. 32, No. 11 (1987).
3) 石井ほか：スケジューリングの近似解法，オペレーションズリサーチ，Vol. 31, No. 1 (1986).
4) Tate, A.: A Review of Knowledge-Based Planning Technique, *Proc. of 5th Tech. Conf. of BCS*, pp. 89-111 (Dec. 1985).
5) Hayes-Roth, F. et al.: *Building Expert Systems*, Addison-Wesley (1986).
6) Smith, S. F. et al.: Constructing and Maintaining Detailed Production Plans: Investigations into the Development of Knowledge-Based Factory Scheduling System, *AI Magazine*, Vol. 7, No. 4, pp. 45-61 (Fall 1986).
7) 中村ほか：ルールベースに基づいたフレキシブル生産システムのスケジューリング，計測自動制御学会論文集，Vol. 23, No. 1, pp. 66-71 (1987).
8) Bruno, G. et al.: A Rule-Based System to Schedule Production, *IEEE Comput.*, Vol. 19, No. 7, pp. 32-40 (July 1986).
9) Goldstein, I. P. et al.: NUDGE, A Knowledge-Based Scheduling Program, *5th IJCAI*, pp. 257-263 (1977).
10) Grant, T. J.: Lessons for OR from A. I.; A Scheduling Case Study, *J. Oper. Soc.*, Vol. 37, No. 1, pp. 41-57 (1986).
11) 関根：線型計画法入門，日本生産性本部(1966).

(昭和63年5月23日受付)
(平成元年6月13日採録)



栗原 謙三（正会員）

昭和23年7月2日生．昭和47年3月早稲田大学理工学部機械工学科卒業．昭和49年3月東京大学大学院工学系研究科修士課程修了．同年4月，(株)日立製作所に入社し，システム開発研究所に勤務．現在，同研究所主任研究員．生産システムの計画，管理技法に関する研究に従事．日本機械学会，電気学会，計測自動制御学会，IEEE各会員．

原 敬市（正会員）

昭和32年生．昭和56年東京理科大学理工学部工業化学科卒業．同年4月日立マイクロコンピュータエンジニアリング(株)入社．以来，(株)日立製作所システム開発研究所にて，AI 技術を応用した計画システムの研究開発に従事し，現在に至る．

小林 隆（正会員）

昭和31年生．昭和55年早稲田大学理工学部機械工学科卒業．昭和57年同大大学院修士課程修了．同年，(株)日立製作所入社．システム開発研究所において，知識工学応用システム，計算機周辺装置の制御などの研究に従事．電気学会，計測自動制御学会，IEEE 各会員．

汐見 龍徳（正会員）

昭和27年生．昭和45年国立熊本電波高校卒業．同年4月(株)日立製作所入社．情報システム工場に勤務し，現在に至る．大型計算機を中心としたシステムの設計に従事．