

確率的局所探索と PERT を組合せた 駅構内入換計画作成アルゴリズム

富井 規雄 周 利剣 福村 直登

〒185-0035 国分寺市光町 2-8-38 (財) 鉄道総合技術研究所
TEL: (042)573-7311 FAX: (042)573-7311
email: tomii@rttri.or.jp

駅構内入換計画作成問題とは、鉄道の駅における車両の入換スケジューリングを作成する問題である。この問題は、ある種の有限資源プロジェクトスケジューリング問題(RCPSP)と見なすことができる。しかしながら、一般の RCPSP と比較すると、プロジェクトを構成する作業の数を問題を解く過程で動的に変化させる必要があること、作業の実行時刻に強い制約があること、プロジェクトの数が多いこと、等の特徴がある。本稿では、駅構内入換計画作成問題に対して、確率的局所探索と PERT を組み合わせたアルゴリズムを提唱する。あわせて、このアルゴリズムを実際規模の問題に適用した場合の結果について報告する。

An Algorithm for Shunting Scheduling Problems combining Probabilistic Local Search and PERT

Norio TOMII Li Jian ZHOU Naoto FUKUMURA
Railway Technical Research Institute

Shunting scheduling problems in railway stations can be regarded as a sort of resource constrained project scheduling problems (RCPSs). But unlike normal RCPSs, shunting scheduling problems have such requirements that the number of work which consists of a project have to be dynamically changed in the process of solving the problem, some of the work have to be performed at prescribed timing etc. We propose an efficient algorithm for shunting scheduling problems combining probabilistic local search and PERT. Local search and PERT is combined so that the candidates for answers in the local search process are evaluated by PERT. This enables to reduce the search space of the local search to a great extent and to make the algorithm work quite fast.

1. はじめに

鉄道においては、輸送形態の複雑化や、コスト削減への対応を主な目的として、列車ダイヤを始めとする輸送計画作成作業を効率化することが強く求められている。本稿で対象とする駅構内での車両入換のスケジュール作成作業についても、従来の人手に依存した形態を計算機を主体とした進め方に移行することが強く求められている。そのためには、駅ごとに異なる設備や列車ダイヤの条件、計画作成上の知識などを柔軟に取り込むことのできる、拡張性に富んだ、汎用的なスケジューリングアルゴリズムが必要となる。

駅構内入換計画（以下、構内入換計画）とは、駅における車両の入換作業のスケジュールのことを言う。すなわち、列車を引上線に移動させる作業が存在する駅において、どの列車を、どの引上線に、いつ引上げるかを定めた計画が構内入換計画である。構内入換計画は、ダイヤ改正時に作成される列車ダイヤ（基本ダイヤ）や、臨時列車が運転される場合の列車ダイヤ（実

施ダイヤ）の作成とあわせて作成される。

従来、構内入換計画は、人手で作成されてきており、自動作成に関する研究は報告されていない。そのために、作成には多大な労力と時間を要しているのが実状である。現に、ある駅では、1度のダイヤ改正のために、延べ 540 時間も時間を費やしているという報告があり[1]、これの作成を迅速に行なうことができれば、ダイヤ改正作業に要する時間の短縮等の点で大きな効果がある。

構内入換計画は、鉄道においてきわめて重要な位置を占める[2]。すなわち、列車ダイヤにおいて列車の到着・出発の時刻や番線が定められているが、列車ダイヤを物理的に実現可能とするためには、車両がその時刻にその番線に出現可能でなければならない。そして、それを規定したものが構内入換計画だからである。

従って、構内入換計画の第一の目的は、列車ダイヤで定められた列車の着発時刻と番線を条件とした構内入換計画が作成可能であることを確認することにある。

しかしながら、列車ダイヤは必ずしも構内入換計画を念頭において作成されるわけではないため、列車ダイヤをすべて守った構内入換計画が作成可能であるという保証はない。そのような計画が作成不可能であることが判明した場合、列車ダイヤの修正を行なうことになる。しかし、列車ダイヤはその駅だけでなく、線区全体、場合によっては、他線区の列車との接続等も考慮して作成されるものであり、ある一個所の変更が広範囲に影響を及ぼすこともある。よって、列車ダイヤの変更はなるべく少ないことが望まれる。すなわち、構内入換計画としては、極力列車ダイヤを守った計画を作成する必要がある。

構内入換計画作成時に守るべき制約としては、列車ダイヤの他に、引上線で車両が競合しないこと、引上線への移動時間を守ること、そこへ到達する進路が存在する引上線を使用すること、保安上の理由により列車相互の時間間隔を確保すること、などがある。

構内入換計画問題は、車両の引上線の占有を「作業」、列車の、駅への着から発までの番線の占有と移動を1つの「プロジェクト」、番線と引上線を「資源」と見ることによって、ある種の有限資源スケジューリング問題 (Resource Constrained Project Scheduling Problem, 以下、RCPPSP と呼ぶ。)[3]ととらえることができる。

近年、スケジューリング問題に対して、シミュレイトイドアニーリング、タブーサーチ、遺伝的局所探索等の局所探索の有効性が注目されている[4][5]。例えば、RCPPSP の特別な場合であるジョブショップスケジューリング問題 (JSSP) に対しても、局所探索が有効であることが示されており[6][7]、本稿でも、駅構内入換計画を局所探索を用いて解くことを検討する。

一般に、局所探索を用いたアルゴリズムを考える場合、近傍をどのように定めるか、および近傍内でどのような探索を行うかが重要である。

しかしながら、構内入換計画には、2節で詳述するように、問題を解く過程において問題自体を動的に変化させざるを得ないこと、作業の順序だけでなく作業の実行時刻までを決定する必要があるため、探索空間が大きくなること、などの通常のスケジューリング問題にはない特徴がある。従って、従来の近傍空間の考え方や探索アルゴリズムをそのまま用いることはできない。

本稿では、構内入換計画を、資源の割当て問題と資源の使用時刻決定問題の2つに分離する。そして、前者を解くためには、作業実行時刻を陽に考えない探索空間の中で局所探索を実施し、そこで得られた解候補に対して、作業実行時刻を決定して解候補の評価を行なう、という二段階の探索手法を提案する。アルゴリズムの概要は次の通りである。

1. 構内入換計画のある種の PERT ネットワーク (入換計画ネットワーク) で表現する。
2. 入換計画ネットワークを変形するオペレータを定義し、それを用いて、近傍空間 (1-オペレータ近傍)

を定義する。

3. 1-オペレータ近傍内の局所探索を行なう。ただし、近傍内の全探索を行なうのではなく、制約を違反した箇所に着目した効率のよい探索を行なう。
 4. 局所探索で見出された複数の解候補に対して、PERT の計算を実施し、制約を満たすかどうかを評価する。
 5. 解候補の中からルーレット選択によって、1つの解候補を選択し、それに対する局所探索を繰り返す。
- これにより、局所探索の探索空間を小規模にすることに成功し、アルゴリズムの効率が向上する。

以下、2節では、構内入換計画の概要とスケジューリング問題としての特徴を述べる。3節では、構内入換計画に対する確率的局所近傍探索と PERT を組み合わせたアルゴリズムを導入する。そして、4節では、現実の列車ダイヤを用いた実験結果を紹介し、本アルゴリズムが良好な結果を与えることを実証する。

2. 駅構内入換計画

2.1 駅構内入換計画とは

図1に示すような配線を持つ駅に対する入換計画の例を図2に示す。この図は作業ダイヤ図と呼ばれ、横軸方向の時間の経過にともなう車両の移動スケジュールを表現している。

図1の配線において、番線1から番線3は、列車が駅外から直接到着できる箇所 (番線) である。それに対して、上り引上線、下り引上線は、到着した列車が折返しになる場合などに、一時的にその車両を留置するために使用される箇所 (引上線) である。番線4は、ホームがないために列車の着発線としては使用されず、やはり、車両の一時留置のために使用される番線で、留置用番線と呼ぶ。引上線と留置用番線は、いずれも入換のために使用される箇所であるので、本稿ではこれら両方をまとめて、引上線と呼ぶ。

車両を、番線と引上線相互間、あるいは、引上線相互間で移動させることを、入換と呼ぶ。到着列車の着番線と発列車の発番線が異なる場合や、着番線と発番線が同じであっても、着時刻と発時刻の間にその番線を他の列車が使用する場合には、入換が必要となる。

さらに、一旦引上線に移動させた列車であっても、その引上線を他の列車が使用せざるを得ない場合 (他に空いている引上線がない等の理由による) には、更に別の引上線に入換を行なう。これを2段入換と呼ぶ。

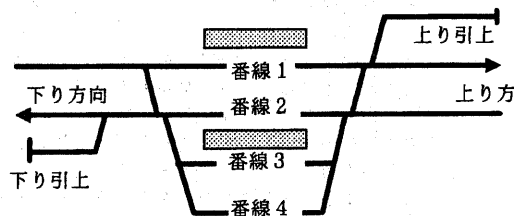


図1: 駅の配線の例

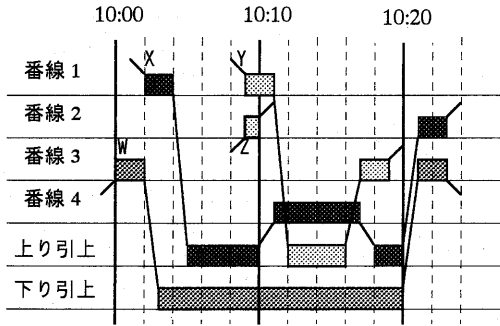


図 2：入換計画の例

列車の駅への到着時刻、発車時刻、到着番線、発車番線は、あらかじめ、列車ダイヤとして定められており（列車ダイヤで定められたこれらの時刻、番線をそれぞれ、計画着時刻、計画発時刻、計画着番線、計画発番線と呼ぶ。またこれらを総称して、それぞれ、計画時刻、計画番線と呼ぶことがある。）、それを条件として、構内入換計画を作成することになる。具体的には、入換が必要である列車に対して、

- ・どの引上線に入換を行なうか
 - ・それぞれの入換に対する、入換の実行時刻（番線から引上線への発時刻・着時刻、引上線から番線への発時刻・着時刻）
- を決定することが、構内入換計画作成の目的である。

2.2 構内入換計画の制約

構内入換計画作成にあたっては、次のような制約を考慮しなければならない。

(1)列車ダイヤに関する制約

- ・列車ダイヤで定められた計画番線、計画時刻を守らなければならない。

(2)設備に起因する制約

- ・進路の有無：移動前の地点と移動後の地点の間に物理的な経路が存在しなければならない。
- ・引上線の有効長：引上線の長さには、制限があるため、収容できる列車の両数には、限りがある。
- ・引上線の競合：通常は、1つの引上線には、1編成の車両しか収容しない。

(3)時間に関する制約

- ・番線、引上線での停車時間：番線や引上線では、乗客の乗降や車内点検のために、ある定められた時間以上の停車時間（最小停車時間）を確保しなければならない。
- ・入換時間：番線と引上線との間の移動時間（入換時間）は、移動の経路ごとに定められているある一定値（駅によって、場所によって異なるが、おおむね2分程度）でなければならない。これは、この時間が短かすぎると物理的に到着不可能であり、長すぎるとその進路を支障する時間が長くな

るためである。

- ・進路支障時間：鉄道の駅においては、複数の進路が交差する場合がある。図 1 においては、番線 1 から上り方向に出発する進路と、上り引上から番線 2 への進路が互いに交差している。このような場合、これらの進路上での入換を同時に実行することはできない。それらの進路上での入換は、分岐器が転換する時間（これを進路支障時間と呼ぶ）だけの間隔をおいて実施しなければならない。これは、保安上、絶対に守らなければならない制約である。入換時刻の決定に際しては、進路支障時間の考慮はきわめて重要である。駅外へ出発する列車や、駅に到着する列車に対して、入換による進路支障が発生すると、それらの列車は、計画時刻を守れなくなるからである。

2.3 構内入換計画の評価基準

前節で述べた制約のうち、列車ダイヤと入換時間以外の制約は、物理的な条件に起因するため、緩和不可能である。列車ダイヤについては、守れるにこしたことはないが、どうしても守れない場合には、その修正件数は少ない方が望ましい。入換時間については、長くなる方に限り、緩和可能ではあるが、もちろん、あまり長くなることは好ましくない。また、不要な入換は、エネルギーの無駄であるので、入換回数は、なるべく少ない方がよい。

以上まとめると、構内入換計画に対する評価基準は、

- ・極力、計画着発時刻を守っていること。
 - ・極力、各入換の入換時間があらかじめ定められた値に一致していること。
 - ・入換回数なるべく少ないこと。
- ということになる。

2.4 有限資源プロジェクトスケジューリング問題としての駅構内入換計画

駅構内入換計画問題は、ある種の有限資源プロジェクトスケジューリング問題と考えることができる。

しかしながら、通常扱われる有限資源プロジェクトスケジューリング問題と比較して、駅構内入換計画は、次のような特徴を持つ。

- (1)プロジェクトを構成する作業の数が所与ではなく、問題解決の過程において、これを動的に変化させる必要がある。これは、2段入換を考慮する必要があることに起因する。
- (2)解として、作業の順序だけでなく、実行時刻までを決定する必要がある。これは、計画時刻を守らなければならないことに起因する。
- (3)通常の RCPSP では、作業と作業の間の時間間隔については、制約がない。しかし、構内入換計画問題では、作業と作業の間の時間間隔をある一定の値にしなければならないケースが存在する。これは、入換時間がある定められた値にしなければならない

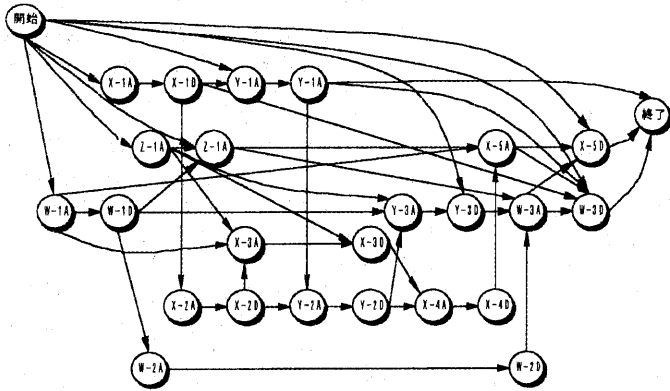


図4：入換計画ネットワークの例

いことに起因する。

- (4)対象プロジェクトの数が多し。駅の規模によって異なるが、1日分では、数百本から1000本程度の列車が関係する。

3. 遺伝的局所探索と PERT を組み合わせた駅構内入換計画作成アルゴリズム

3.1 基本的考え方

一見制約を満たしていない入換計画でも、入換時刻を調整することにより、制約を満たす入換計画を得ることができる場合がある。例えば、図3に示す入換計画は、引上線で車両が競合しているため、このままでは、制約を満たしていない。しかし、引上番線から番線2への入換時刻を早め、番線3から引上番線への入換時刻を遅くすることによって、競合を解消することができる。つまり、時刻以外の引上線の使用計画は同一であっても、使用時刻を変えるだけで結果が制約を満たしたり、満たさなかったりする。

このことは、構内入換計画では、各作業の順序だけではなく、各作業の実行時刻を制約を満たすように決定する必要があることを意味する。しかし、入換時刻

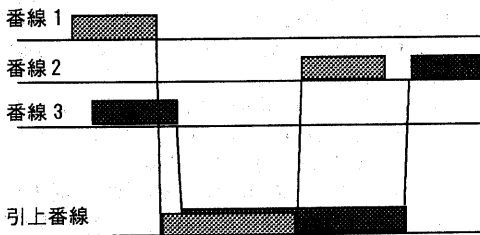


図3：引上線で車両が競合する入換計画

の決定までを探索の中に組み入れると探索空間が非常に大きくなり、探索効率の面で問題がある。

そこで、本稿では、時刻を考慮しない引上線の使用計画を仮に決定した後、その条件のもとで、制約を満たすように入換実行時刻を決定することができるかを探索する、という2段階の探索手法を提唱する。

このアルゴリズムの全体構成を、手続き1に示し、各ステップの詳細を以降の節に示す。

Step0 初期設定：ヒューリスティクスによって初期解 x を求め、 x を暫定解とする。 $k=0$ とする。

Step1 近傍探索： x の近傍内の探索を行ない、解候補 x_1, x_2, \dots, x_m を生成する。 $k:=k+1$ とする。

Step2 評価： x_1, x_2, \dots, x_m に対する評価値を計算する。その中に解があれば、それを出力しておわり。 x_1, x_2, \dots, x_m の中に x よりもよい解があれば、それを暫定解とする。

Step3 選択： x_1, x_2, \dots, x_m の中から1つを選択し、それを x とし、Step1へ。ただし、 $k>L$ の時は、その時点での暫定解を x として、Step1へ。

手続き1：アルゴリズムの全体構成

3.2 初期設定

ヒューリスティクスによって入換番線と入換時刻を決定し、初期解を1つ生成する。ここで用いるヒューリスティクスは、ごく単純なもので、

- ・計画着番線と計画発番線の組合せから使用するべき引上線を決める。

- ・入換時刻は、最小停車時分のみから決定する。

というものである。つまり、この段階では引上線での車両の競合に関しては一切考慮していない。

3.3 近傍探索

3.3.1 構内入換計画の PERT 表現

構内入換計画を、次のようにして、PERT ネットワークで表現する。これを、入換計画ネットワークと称する。入換計画ネットワークは、 (N, A) で表わされる。ここで、 N は、ノードの集合、 A は、アークの集合で、 $A \subset N \times N$ である。アークには、重み $w \geq 0$ が付されている。この意味は、次の通りである。

- ・ノードは、車両の番線・引上線の着事象または発事象を表わす(それぞれ、着ノード、発ノードと言う)。

・アークは、ノード間の時間的実行順序を表わす。アークの重みは、始点ノードの事象と終点ノードの事象との間に最低限必要な時間を示す。

アークには、表 1 に示す 6 種類が存在する。入換計画ネットワークは、その時点での解候補が持つ入換時刻をもとにして決定された作業実行順序に従って構成される。図 2 に対する入換計画ネットワークを図 4 に示す。(図 4 のノード中の文字は、列車・ノード番号・着発区分を示す。着発区分は、A が着、D が発である。)

3.3.2 入換可能時刻計算

解候補の入換ネットワークに対して、各ノードの実行可能時刻を計算する。計算方法は、通常の PERT の計算とほぼ同様であるが、通常の計算方法そのままでは、入換時間が守られない場合があるため、入換アークに対しては、バックトラックを行なって、入換時間を守るような結果を得るようにしている。これにより、設備に関する制約、入換時間以外の時間に関する制約をすべて守った計算結果が得られる。

表 1: アークの種類

種別	内容	重み
運用	ある列車の番線占有の着ノードと発ノードの間	最小停車時間
入換	異なる番線・引上線間の発ノードと、着ノードの間	入換時間
番線	同一番線・引上線上の発ノードと着ノードの間	続行時隔
進路支障	支障する進路を使用するノード間	進路支障時間
計画時刻	開始ノードと、計画時刻をもつノードの間	計画時刻
終了	番線・引上線上の最後のノードと、終了ノードの間	0

3.3.3 1-オペレータ近傍

JSSP に対する近傍探索を行なう場合、近傍として、クリティカルブロック上の隣接する作業の処理順序の入換による近傍[8][9]、クリティカルブロック内の作業をそのブロックの先頭、あるいは最後への移動による近傍[10][11]などが用いられている。しかし、構内入換計画においては、2段入換を考慮しなければならないことから、問題自体が動的に変化する。従って、単純な作業順序の入換だけでは対処できない。そこで、本稿では、入換計画ネットワークの変形操作による近傍という考え方を導入する。

入換計画ネットワークに対するネットワーク変形オペレータ (以下、変形オペレータ) を、表 2 のように定義する。

これを用いて、入換計画ネットワーク P に対する 1-オペレータ近傍 $Nb(P)$ を次のように定義する。

$Nb(P) = \{P \text{ に対して、変形オペレータを 1 回適用して}$

得られる入換計画ネットワーク}

表 2: ネットワーク変形オペレータ

オペレータ	操作の内容
引上線変更	引上番線の変更
2段入換	2段入換の設定
2段入換解消	2段入換の解消
順序変更	隣りあう 2 つの占有の順序変更
進路支障	進路支障支障アークの向きを反転

3.3.4 近傍内の探索

近傍内での探索を行ない、解候補 (複数) を生成する。ただし、ここでは、1-オペレータ近傍内すべてを探索するわけではなく、解を導出する可能性の高い部分のみを探索する。詳細を手続き 2 に示す。

Step1: ノードの選択: 解候補の入換計画ネットワークの中で、計画時刻を守っていないノード、入換時間を守っていないアークの終点のノードから、1 つをランダムに選択する。

Step2: ネットワークの変形

- ・計画時刻を守っていないノードの場合: 計画時刻を守っていないノードの集合を X 、計画時刻を守っているノードの集合を Y とする。ノード $x \in X$ から、クリティカルパスを逆にたどって最初にてあったノード $y \in Y$ に至るパスの集合を、 $Cr(x)$ とする。 $Cr(x)$ の構成要素であるアーク 1 つずつに対して、そのアークの種別ごとに表 3 に示す変形オペレータを適用することにより、それぞれ一つの解候補を得る。
- ・入換時間を守っていないノードの場合: 入換時間を守っていないアークの両端のノードに対して、引上線変更オペレータを適用して、解候補を生成する。

手続き 2: 近傍内の探索

表 3: アーク種別ごとのネットワーク変形オペレータ

アーク種別	適用オペレータ
運用アーク	引上線変更, 2段入換
入換アーク	引上線変更
番線アーク	順序変更
進路支障アーク	進路支障

3.4 評価

計画時刻を守っていないノードの数、入換時間を守っていないアークの数、入換回数観点から解の評価を行なう。評価値は、次の選択ステップで用いる。

3.5 選択

近傍内で探索された複数の解候補から、1 つの解候補を選択する。選択の方法は、評価値をもとにしたル

ーレット選択による。

4. 実行結果と考察

4.1 実行結果

現実のダイヤデータを用いて試行を行なった。対象とした駅は、典型的な中規模の駅で、番線数6、留置用番線数2、引上線数2、合流する線区の数3である。対象時間帯は、この駅の最繁時間帯である、6:00~10:00で、対象となった列車本数は、55本である。

この駅に対する現実のダイヤを入力とした場合の10回の試行結果を表4に示す。探索数は、探索の対象となった解候補の数、Stepは、手続き1のStep1~Step3を何回繰り返したかを示す。また、計画時刻は、計画時刻を守っていない列車の着発の数、入換時間は、入換時間を守っていない入換の回数、入換回数は、入換の回数を示す。なお、手続き1におけるLは15としてプログラムを実行した。

10回の試行において、いずれも、実用とするに足る解を得ることに成功している。また、おおむね数分で、これらの解は得られた(Pentium 166MHz, メモリ 80MB)。

表4：実験結果

試行	探索数	Step	計画時刻	入換時間	入換回数
1	40	9	0	0	28
2	21	7	0	0	28
3	21	7	0	0	28
4	73	17	0	0	28
5	26	8	0	0	28
6	26	8	0	0	28
7	31	9	0	0	28
8	42	9	0	0	28
9	90	17	0	0	28
10	33	9	0	0	28

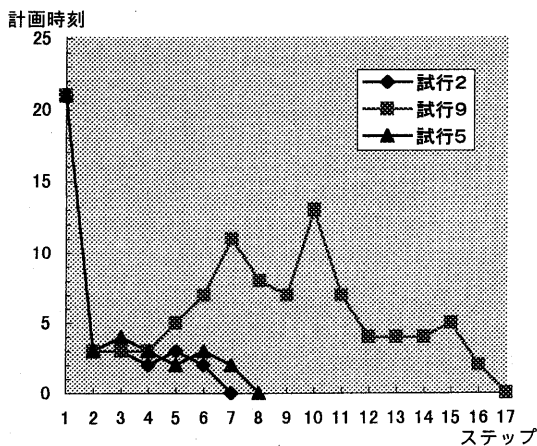


図5：ステップごとの解候補の評価値の推移

4.2 考察

- (1) 試行4と試行9においては、L=15ステップ内では、制約をすべて満たした解が発見されず、暫定解に戻って探索を再開しているが、その後はいずれも2ステップで解を得ている。この事は、一定のステップまで探索を続けた後、暫定解に戻って探索を再開するという戦略の有効性を立証している。
- (2) 図5に、最短で解が発見された試行2、最長であった試行9、その中間である試行5について、各ステップごとに選択された解候補中での計画時刻を守れないノードの数の推移を示す。このグラフから、最良の解候補を選択することが必ずしも得策でないことが分かり、次ステップに渡す解候補をルーレット選択で選定していることの有効性を立証している。

5. おわりに

駅構内入換計画作成問題に対して、確率的局所探索とPERTを組み合わせたアルゴリズムを提唱した。

本アルゴリズムでは、時刻の要素を除いた番線使用計画を局所探索の探索空間とし、得られた解候補に対してPERTを用いて時刻の決定を行なうという、2段階の探索を実施している。また、局所探索のための近傍の定義には、PERTネットワークに対する変形オペレータを用いる考え方を導入した。あわせて近傍内の探索には、クリティカルパス等に着眼した効率のよい方法を提唱した。その結果、効率的なアルゴリズムを構築することが可能になり、実用規模の列車ダイヤデータに対して、実用的な解が得られることを実証した。

参考文献

- [1] 西尾孝義他, データベース利用で自動作成・出力ダイヤ改正作業効率化の研究, JRTR - JR Central Technical Report, No.54.(1997).
- [2] 藤田, 富井他, 鉄道とコンピュータ, 共立出版 (1998).
- [3] 鍋島, スケジューリング理論, 森北出版 (1974).
- [4] 吉川, 制約最適化技術のスケジューリング問題への応用, 人工知能学会誌, Vol.13, No.3 (1998).
- [5] 茨木, 組合せ最適化とスケジューリング問題: 新解法とその動向, 計測自動制御学会誌, Vol.34, No.5, (1995).
- [6] 山田, 中野, 遺伝的局所探索法によるジョブショップスケジューリング問題の解法, 情報処理学会論文誌, Vol.38, No.6 (1997).
- [7] 山田, 中野, 確率的探索と確定的探索の組合せによるジョブショップスケジューリング問題の解法, 情報処理学会論文誌, Vol.37, No.4 (1996).
- [8] Taillard, E. Parallel Taboo Search Techniques for the Job-shop Scheduling Problem, *ORSA, J. on Comput.*, Vol.6, No.2 (1994).
- [9] Laarhoven, P.v., Aarts, E. and Lenstra, J.: Job Shop Scheduling by Simulated Annealing, *Oper. Res.*, Vol.40, No.1, (1992).
- [10] Dell'Amico, M. and Trubian, M.: Applying Tabu Search to the Job-shop Scheduling Problem, *Annals of OR*, Vol.41 (1993).
- [11] Brucker, P., Jurisch, B. and Sievers, B.: A Branch Bound Algorithm for the Job-Shop Scheduling Problem, *Discrete Applied Mathematics*, Vol.49 (1994).