

生産計画立案自動化、最適化技術確立

Establishment of Automation and Optimization Technologies for Production Planning

山根 有輝也^{*1}
Yukiya Yamane

*1 ITデジタル推進部

要旨 鑄造ラインを対象に、生産計画の立案業務を自動化する多目的最適化ツールを開発した。本稿では、業務フローの分析からアルゴリズムの選定、数理モデルの構築、そして実証に至る2年間の取組みについて報告する。混合整数計画法(MIP, Mixed Integer Programming)の数理モデルに対し、熟練担当者の意思決定を模倣する「離散的なペナルティ」を導入し、計画立案工数を63%削減するとともに、段取り替え回数・在庫数・残業時間の3指標を同時に最適化した。

キーワード: 生産計画、鑄造、段取り替え最小化、オペレーションズ・リサーチ、組合せ最適化、混合整数計画法

Abstract A multi-objective optimization tool was developed to automate the production planning process for casting lines. This paper reports on the two-year effort, covering the analysis of workflows, the selection of algorithms, the construction of mathematical models, and the validation results. By incorporating a "discrete penalty" approach to model the decision-making of experienced planners into a Mixed Integer Programming (MIP) mathematical model, we achieved a 63% reduction in planning workload while simultaneously optimizing three metrics: setup change frequency, inventory levels, and overtime hours.

Keywords: Production Planning, Casting, Die Change Minimization, Operations Research, Combinatorial Optimization, Mixed Integer Programming

1 はじめに

1.1 生産計画立案業務の課題

本取組みの対象である鑄造ラインでは、約60基の鑄造設備(以下、マシンと表記)と、1品番につき複数ある金型との組合せで、最大400品番超の製品を製造している。生産計画の立案および修正作業は、従来、熟練の生産管理担当者が多大な時間をかけて手作業で行っていた。事前調査の結果、計画を構成する要素の組合せ数は最大で10の240万乗に達することが判明した。この想像しがたい規模の問題を、自動化を目的とした市販のパッケージソフトウェアで解くのは極めて困難である。

これらの課題に対し、2年間の取組みを経て、熟練担当者のノウハウを取り入れた独自の数理モデルを構築した。このモデルを活用することで、日次計画立案の自動化および最適化を目指すとともに、他工程で最適化技術を活用する際の新たな可能性についても検討した。

1.2 計画立案の業務フロー

月初に暫定的な1か月分の日次計画を2日間かけて作成したうえで、①突発的な変更やトラブルに対応しながら、②稼働情報の登録・確認を踏まえて、③毎日計画の修正を行う。そのため、計画の立案・修正工数は、生産管理の関係部署の累計で月に約1,000時間にも及んでいた(図1)。

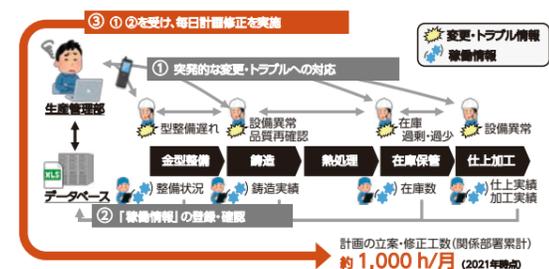


図1 日次計画修正フローと月間工数
Fig.1 Daily Planning Revision Flow and Monthly Workload

1.3 計画立案時に参照するデータ

熟練担当者が計画立案時に参照するデータを表

1に示す。なお、表内の★印は日々の変動が大きい値である:

表1 計画立案時に参照するデータの詳細
Table1 Details of Data Referenced During Planning

データファイル名	値の名称
品番マスタ	★現在の在庫数、★在庫の引き取り数、在庫数の基準値、サイクルタイム、ロット数
マシン割当計画	各品番の生産が割り当てられているマシンの番号
稼働カレンダー	日々の残業時間上限(定時退場日など)、所定休日、休日出勤日

2 数理モデルの構築

2.1 アルゴリズム選定までの試行錯誤

「手持ちの解法にこだわるのではなく、問題にあった解法を探せ」という格言がある通り、生産計画を数学的な問題に変換して解く際には、実務との整合性を保ちながら適切なアルゴリズムを選定することが重要である^[1]。

アルゴリズムの選定にあたり、とくに本取組みの初年度は課題の本質を明らかにするために、あらゆる手段を用いて熟練担当者へのヒアリングを行った。例えば、計画立案作業のPC画面をビデオ通話ツールで録画し、作業の流れや工数の内訳を詳細に解析した。また、熟練担当者のPC作業を背後から直接観察させてもらいながらヒアリングを実施することもあった。さらに、開発者自身が計画立案を試み、その結果を熟練担当者に添削してもらい、具体的な判断基準や思考プロセスを体感的に学んだ。

以上を経て、アルゴリズムの選定と、初期の数理モデルの構築に成功した。なお、技術的な詳細は後述する。

2.2 制約抽出・数理モデル改良サイクルの確立

次年度では、その数理モデルを改良するため、計算結果と実績の差異分析を行った。また、その差異に対する熟練担当者の意見も反映させた。また、マシンや自動搬送装置(AGV)を現地・現物で確認し、ヒアリングで得た制約の背景を調査することで、さらなる改良を図った。

そして、これらの活動を標準化し、制約抽出・数理モデルの改良サイクルを確立した(図2)。

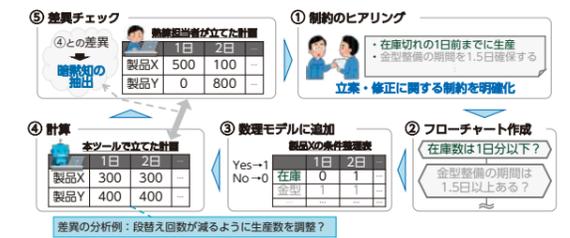


図2 熟練担当者へのヒアリングと差異分析による制約抽出・数理モデル改良サイクル
Fig.2 Constraint Extraction and Mathematical Model Refinement Cycle via Expert Interviews and Gap Analysis

2.3 混合整数計画法(MIP)の採用

本取組みでは、混合整数計画法(MIP)というアルゴリズムを採用し、日次の生産計画に関する制約や条件などを線形の数式のみを用いて定式化した。そして、生産数、段取り替え(段替え)の回数、在庫数などを同時に最適化するツールを開発した。

選定にあたり、「近似解法」という種類のアルゴリズム群も検討したものの、制約が多すぎて対応できない可能性がある点が課題となった。一方で、今回採用したMIPはすべての制約を満たす「解」を出力できる(「解」は、計算の結果得られる計画案を示す)。また、MIPは数理モデルが構築できれば、その計算はソルバ(solver)とよばれるソフトウェアに任せられるため、他のアルゴリズムと比べて習得難易度が比較的低いという利点もある。

MIPの数理モデルは、決定変数、制約式、目的関数の3要素からなる(図3)。以下では各要素について説明する。

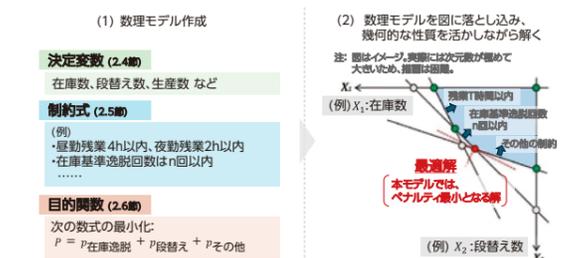


図3 MIPの数理モデルにおける3要素(決定変数・制約式・目的関数)の概念図
Fig.3 Conceptual Diagram of the Three Components in the MIP Model—Decision Variables, Constraints, and Objective Function

2.4 決定変数

決定変数とは、最適化の対象となる変数のことである。本モデルでは、各マシン m が品番 i を、稼働日 d にいくつ生産するかを表す変数 $x_{m,i,d}$ を中心と

し、その生産数を昼勤・夜勤・休日出勤というシフト(直)ごとに加え、1時間単位に細分化して管理する。この仕組みにより、熟練担当者の考え方と同様の粒度での計画立案が実現できる。

また、在庫数については欠品や基準逸脱(過剰・過少)などの状態を判定し、ペナルティとよばれる罰則点を自動で加算するための変数(ペナルティ変数)も設けている。さらに、週内の負荷のピークを管理する変数を設けて平準化を図った。

2.5 制約式

制約式とは、生産計画を立案する際のルールや条件を数学的に表現したものである。前章の表1に示した参照データをもとに、鑄造ラインの制約や条件を整理することで設定した。各データファイルと制約式の対応を表2に示す:

表2 各データファイルと制約式の対応
Table 2 Correspondence Between Data Files and Constraint Formulations

データファイル名	設定した制約式
品番マスタ	在庫数の推移制約(本節で説明) 在庫基準逸脱の回数制限制約
マシン割当計画	品番×マシンの組合せ制約 生産能力および段替え回数の制約
稼働カレンダー	残業時間上限、休日制約 週内の負荷バラつき防止制約

さらに本取組みでは追加制約として、需給バランスや金型の整備間隔などに関する熟練担当者のノウハウを取り入れることで、実務に即した計画立案を可能にしている。

ここで、日ごとの在庫数の推移を表現する制約式を紹介する。品番*i*について、稼働日*d*の翌日の在庫数 $Z_{i,d+1}$ は以下の制約式(1)で表せる:

$$Z_{i,d+1} = Z_{i,d} + \sum_m x_{m,i,d} - h_{i,d} \quad (1)$$

(各変数の説明)

・ $Z_{i,d}$: 品番*i*の稼働日*d*時点の在庫数

・ $h_{i,d}$: 品番*i*の稼働日*d*での引き取り数

なおこの制約式は、複数のマシンが分担して同じ品番を生産するケース(いわゆるブリッジ生産)にも対応できる。

2.6 目的関数

目的関数とは、最小化(または最大化)の対象となる関数である。本モデルでは、計画のペナルティを最小化させるように設定している。ペナルティは、前述のペナルティ変数に、優先順位を示す係数(ペナルティ係数)を乗じた数の和で表される。ペナルティ係数は優先順位が高いほうから、在庫基準逸脱、段替え、負荷変動などについて設定している。詳細は次の章で説明する。

3 ペナルティの仕組み

3.1 離散的なペナルティの採用

ペナルティの決定にあたっては、各項目が同時に最適化できるよう、熟練担当者へのヒアリングや実験を重ねた。そして、その過程から得られた知見をもとに、評価指標の目標値からのかい離を区分ごとに捉える「離散的なペナルティ」を考案、採用した^[2]。

一方、既存技術である連続的なペナルティも限定エリアを対象に試行したが、計算時間が延びるという課題があった。これは、連続的なペナルティの最小化の過程で、解の組合せが無数にあるためと考えられる(図4)。

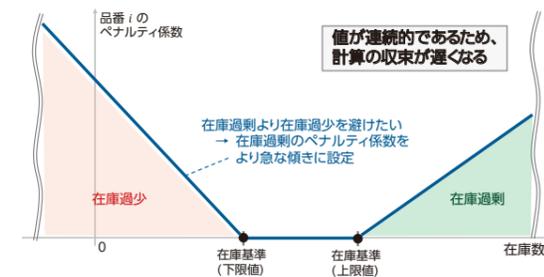


図4 連続的なペナルティ関数のイメージ
Fig.4 Illustration of a Continuous Penalty Function

3.2 在庫基準逸脱ペナルティ

代表例として、在庫基準に対して在庫数が過剰または過少となった場合に適用されるペナルティの仕組みを説明する。通常、在庫基準は上限値と下限値の2つのしきい値で管理されるが、検証の結果、しきい値を段階的に細分化することで計算精度を向上できることが判明した。この知見をもとに、熟練担当者と協議を重ねながらしきい値を調

整し、精度をさらに高めた。

ここからペナルティ算出の流れを示す。まず、品番*i*の在庫基準のうち下限を u_i 、上限を o_i とし、加えて下限の3分の2を t_i 、上限の1.5倍を r_i という定数で表す。

続いて、稼働日*d*、品番*i*について、ペナルティ変数の組 $\delta_{i,d,1}.. \delta_{i,d,4}$ を、式(2) - (5)で定義する(ただし、実際の制約式では、Big-M法を利用)^[3]。また、各ペナルティ変数に対応するペナルティ係数 $k_1.. k_4$ とする(図5)。

$$\begin{cases} \delta_{i,d,1} = 1 & (\text{if } Z_{i,d} < t_i) \\ \delta_{i,d,1} = 0 & (\text{if } t_i \leq Z_{i,d}) \end{cases} \quad (2)$$

$$\begin{cases} \delta_{i,d,2} = 1 & (\text{if } t_i \leq Z_{i,d} < u_i) \\ \delta_{i,d,2} = 0 & (\text{if } Z_{i,d} < t_i \text{ or } u_i \leq Z_{i,d}) \end{cases} \quad (3)$$

$$\begin{cases} \delta_{i,d,3} = 1 & (\text{if } o_i < Z_{i,d} \leq r_i) \\ \delta_{i,d,3} = 0 & (\text{if } Z_{i,d} \leq o_i \text{ or } r_i < Z_{i,d}) \end{cases} \quad (4)$$

$$\begin{cases} \delta_{i,d,4} = 1 & (\text{if } r_i \leq Z_{i,d}) \\ \delta_{i,d,4} = 0 & (\text{if } Z_{i,d} < r_i) \end{cases} \quad (5)$$

このとき、在庫基準逸脱ペナルティ $p_{\text{在庫逸脱}}$ は式(6)のように表せる。このペナルティが最小となるとき、製造部門にとって最適な在庫数となる。

$$P_{\text{在庫逸脱}} = \sum_i \sum_d (k_1 \delta_{i,d,1} + k_2 \delta_{i,d,2} + k_3 \delta_{i,d,3} + k_4 \delta_{i,d,4}) \quad (6)$$

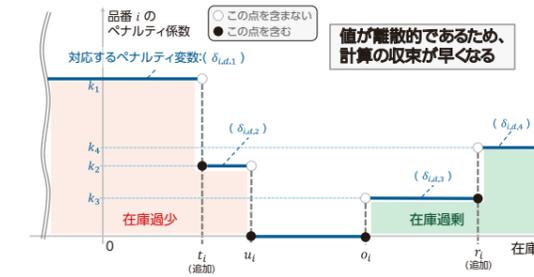


図5 在庫基準逸脱を評価する離散ペナルティの仕組み
Fig.5 Discrete Penalty Mechanism for Inventory Deviation

3.3 目的関数へのペナルティの組み込み

最終的に、すべてのペナルティの和をとると、多目的最適化が可能な目的関数ができあがる。目的関数 P は式(7)となる。

$$P = p_{\text{在庫逸脱}} + p_{\text{段替え}} + p_{\text{その他}} \quad (7)$$

ここで、 $p_{\text{段替え}}$ は、1日あたりの段替え回数に応じ

て発生するペナルティを示す。同じ日に複数の段替えが重なると、生産可能な時間が減少するだけでなく、製造部門の作業者の負担も大きくなる。この問題に対応するため、段替えが重なる日に対して指数関数的に増加するペナルティ係数を設定し、段替え重複を抑制できる仕組みとした。また、 $p_{\text{その他}}$ には負荷の平準化や金型整備期間の確保に関する項目が含まれる。

前述のように各項目のペナルティに適切な係数を設定することで、在庫基準逸脱を回避し、段替えの重複を抑制し、さらに負荷の平準化などを同時に考慮した最適化を実現できる。

4 実証結果と考察

4.1 計画立案の工数および働き方の変化

導入前は熟練作業員のみで計画立案していたため、手作業と電話対応で残業も多かった。さらに、1か月分の暫定計画を立てるのに2日間かかっていた。

導入後は本ツールが解を提示することで、月初の計画立案工数が63%減少し、空いた分を在庫の基準値の見直しなどといった改善活動に回せるようになった(図6)。

また、1か月分の暫定計画を1日で立てられるようになった。これにより、数日後の見通しが目で見てわかるため、経験と勘が頼りであった電話調整の負担も減少した。これにより新任担当者でも業務の質を落とさず作業ができるようになった(図7)。



図6 在庫基準値などを調整できる最適化ツールの詳細設定画面(例)
Fig.6 Example of the Optimization Tool's Detailed Configuration Screen for Revising Inventory Thresholds

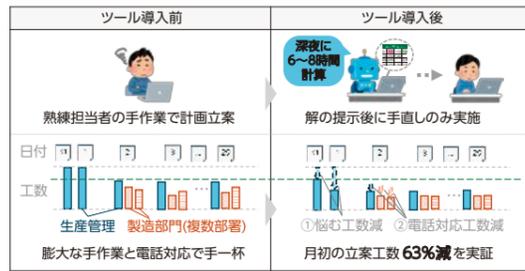


図7 ツール導入前後における計画立案工数と残業時間の削減効果
Fig.7 Reduction of Planning Workload and Overtime by Tool Implementation

4.2 計算時間・精度の比較

2022年の3か月分のデータを用い、複数のソルバについて、計算時間と得られる解の精度を比較した。なお計算環境は、OSとしてWindows (R) 10 Enterprise 64bit(検証当時)、プロセッサとしてIntel(R) Xeon(R) Gold 6226R(2.90 GHz)、RAMとして192 GBを備えたPCである。

比較項目は以下の3点である。

- 1か月分の計画立案問題について解が出せたか
- 2週間分の計画立案問題について解が出せたか
- 得られた解の収束度合い

3点目の「収束」とは、ペナルティが十分に小さい解が得られており、実務で利用できるほどの品質が期待できることを意味する。ソルバの比較結果を表3に示す：

表3 ソルバの比較結果
Table 3 Comparison Results of Solvers

ソルバ	費用	1か月分の計画	2週間分の計画	収束度合い
Gurobi ^[4]	商用	6~8時間で収束	— (評価を省略)	十分に収束
SCIP ^[5]	無償	24時間で未完	約8時間で完了	24時間で収束せず
CBC ^[6]	無償	24時間で未完	24時間で未完	— (評価不可)

Gurobiは商用であるが計算性能が非常に高く、1か月分の計画の計算は6~8時間で収束した。また、1か月単位の生産計画を実用的な時間内に立案できる唯一のソルバとして採用に至った。一方、無償ソルバのSCIP、CBCでは、解を得られる場合はあったものの、24時間計算しても十分に収束しなかった。

4.3 考察

得られた解と実績の差異分析および、離散ペナルティの導入により、熟練担当者の経験からしきい値として用いていた条件(例:在庫の基準値やペナルティ係数など)を数理モデル内で再現できた。また、連続的なペナルティを用いる場合に比べて、計算の高速化を実現した。

実証結果においては、計画立案工数を削減できた一方、欠品や過剰在庫、段替え回数などの評価指標については、従来水準から変化がなかった。各評価指標のさらなる改善を行うには、熟練担当者の評価に基づく数理モデルではなく、原単位を反映したコストなどに置き換えるのも一案である。

また、ソルバの比較結果からは、大規模な最適化問題では商用ソルバが実用的である一方、問題の分割やモデリングの見直しといった工夫を行えば、無償ソルバも活用できることが示された。

以上により、最適化技術を他工程などに適用する際には、熟練担当者と連携しながら評価項目を決めたのちにアルゴリズムを選定し、ソルバへの投資や数理モデルの工夫を検討することが「最適」な戦略であるといえよう。

5 おわりに

5.1 まとめ

本取り組みでは、鑄造ラインの生産計画立案の自動化・最適化を行う数理モデルを構築した。在庫基準逸脱や段替え回数といった複数指標について、独自の離散的なペナルティを活用して同時に考慮できる仕組みを実装した。その成果として計画立案工数の63%削減を実現した。また、ソルバへの投資またはモデルの工夫のどちらを進めるかなど、最適化技術を活用するうえでの汎用的な知見を得た。

5.2 今後の展望

今後の展望として、主に以下の3点を据えている。

まず、複数拠点が連携して生産を行う場合でも全体最適を実現する「多拠点同時最適化」への拡張が課題である。

2点目は、任意の条件で日次や週次の計画を素

早く修正できる「シミュレーション機能」を導入し、需要変動や設備異常への迅速対応を目指す。

3点目は「生成AIの活用」であり、今後、当社ひいては製造業界に最適化技術の活用を広げるうえで欠かせないものと考えられる。自然言語で記述されたルールや熟練担当者のノウハウを定式化することで、数理モデルを自動で調整する仕組みを検討する。これにより、複雑な制約条件を効率よく反映でき、計画立案の精度とお客様への対応の柔軟性がさらに向上するであろう。

以上の展望を踏まえ、今後も理論と実践を組み合わせながら、鑄造ラインのみならず幅広い工程での適用を目指した最適化手法の高度化を進めていく。

謝辞

本取り組みにおける共同研究先の(株)豊田中央研究所の皆様および、ご協力を頂いた社内外の関係者の皆様に心より感謝を申し上げます。

参考文献

- [1]久保幹雄: Python言語による実務で使える100+の最適化問題, 2025/7/16閲覧,
< <https://scmopt.github.io/opt100/> >.
- [2] (株)豊田中央研究所、(株)豊田自動織機:日本公開特許公報, 特開2024-134808, 2024/10/4.
- [3]宮代隆平: 整数最適化アプローチへの入門, IEICE 総合大会講演資料, p.17, 2019/3/22.
< https://web.tuat.ac.jp/~miya/miyashiro_IEICE2019.pdf >.
- [4]Gurobi Optimizer, 2025/7/16閲覧,
< <https://scmopt.github.io/opt100/> >.
- [5]SCIP (Solving Constraint Integer Programs), 2025/7/16閲覧,
< <https://scipopt.org/> >.
- [6]CBC (Coin-or branch and cut), 2025/7/16閲覧,
< <https://github.com/coin-or/Cbc> >.

著者紹介



山根 有輝也