

# スケジュール問題への AI 活用 —計算論的インタフェースとしての AI エージェント—

Y.Matsuda and ChatGPT5.2

2026 年 2 月 2 日

## 1 動機（前提）

ロジスティックや生産計画に代表されるスケジュール問題は、組合せ最適化、線形計画法、最短経路問題など、数学的に良く研究された枠組みを持つ。いったん問題が数式として定式化されれば、高速かつ信頼性の高い既存アルゴリズムを用いることで、理論的には最適解または良質な近似解を得ることができる。しかし現実の運用においては、スケジュール問題解決ツールが継続的に開発され続けていることが示すように、単一の定式化や単一のアルゴリズムでは現実の複雑さを吸収しきれないという事実がある。

この背景には、現実世界における制約や条件が、(i) 事前に完全列挙できない、(ii) 時間とともに変化する、(iii) 暗黙知や慣習、人的判断を含む、という性質を持つことがある。本レポートでは、この問題意識を出発点として、**スケジュールアルゴリズムそのものを高度化するのではなく、複数の既存アルゴリズムをいかに組み合わせ、現実と数理の間を接続するか**という観点から、AI エージェントの役割を整理する。

## 2 議論の枠組み

本レポートの立場は次の一点に集約される。すなわち、スケジュール問題の本質的困難は「解法アルゴリズムの不足」ではなく、「どのような問題として切り出すかが状況ごとに変化する」点にある。したがって、数理最適化アルゴリズム自体は既存のものを活用し、その入力として与える**制約条件、重み、優先度、問題構造**を動的に調整する計算論的インタフェースが必要となる。

このインタフェースとして AI（あるいは統計的機械学習）を位置づけることで、現実の複雑さを直接アルゴリズム内部に埋め込むのではなく、複数のアルゴリズムの**組み合わせ方**によって吸収する構成が可能となる。

## 3 既存のスケジュールアルゴリズムとその役割

ここでは代表的なスケジュール関連アルゴリズムを挙げ、それぞれの「役割」を簡潔に整理する。

### 3.1 線形計画法・整数計画法

線形計画法および整数計画法は、制約条件と目的関数が明示的に定義できる場合に、最適性が保証された解を与える。役割は、**明確に定義可能な制約と目的のもとでの基準解を提供すること**にある。一方で、定式化に含められない条件には対応できない。

### 3.2 最短経路・ネットワーク最適化

最短経路問題やフロー問題は、移動や配送を中心とするスケジュールにおいて中核的な役割を果たす。役割は、**距離・時間・コストといった単一尺度に基づく効率性の最大化**であり、局所的な評価軸に強い。

### 3.3 ヒューリスティクス・メタヒューリスティクス

局所探索、貪欲法、遺伝的アルゴリズムなどは、厳密最適性よりも実行可能性と柔軟性を重視する。役割は、**厳密解法では扱いにくい大規模・非線形問題に対し、実用的な解を迅速に生成すること**にある。

重要なのは、これらはいずれも異なる仮定のもとで問題を切り取るアルゴリズムであり、単体で万能ではないという点である。

## 4 AI エージェントの役割：組み合わせの設計

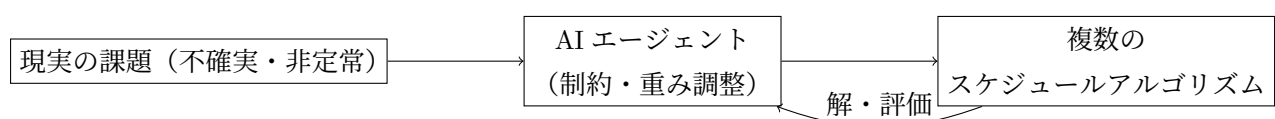
本レポートの主たるテーマは、AI エージェントを**スケジュール問題の解決者**としてではなく、**計算論的インタフェース**として位置づける点にある。AI エージェントの役割は次の三点に整理できる。

- 現実の状況（需要変動、例外事象、人的判断）を観測し、数理モデルに反映可能な形で**制約条件や重みの変更**として表現すること。
- 複数のスケジュールアルゴリズムに対し、同一の現実を異なる定式化で入力し、**並列に解を生成**させること。
- 得られた複数の解を比較・評価し、現実的リスクや運用適合性の観点から**採用・修正・再計算**を制御すること。

ここで重要なのは、各アルゴリズム自体のパラメータを過度に複雑化しない点である。現実の複雑さは、個々のアルゴリズム内部で表現するのではなく、**異なる性質を持つアルゴリズムの組み合わせ**によって吸収される。

## 5 関係構造の模式図

以下に、現実の課題、AI エージェント、スケジュールアルゴリズムの関係を模式的に示す。



## 6 AI エージェントの意思決定原理：何を最小化／最大化するのか

AI エージェントを計算論的インタフェースとして位置づけた場合、その意思決定は単一の目的関数の最適化として記述されない点が重要である。すなわち、AI エージェントは「スケジュールそのもの」を最適化するのではなく、**スケジュール生成プロセス全体の健全性**を対象として評価・制御を行う。

本レポートの立場では、AI エージェントが暗黙的に最小化／最大化している対象は、以下のようなメタレベルの量として整理できる。

### ・最小化対象：

- 現実との乖離リスク（生成されたスケジュールが運用上破綻する確率）
- 再計算・再調整の頻度（現場介入コスト）
- アルゴリズム仮定の偏り（単一モデルへの過度な依存）

### ・最大化対象：

- 複数アルゴリズム解の整合性・合意度
- 制約違反に対する耐性（小さな条件変化への頑健性）
- 人間判断との整合性（説明可能性・納得可能性）

このように、AI エージェントの意思決定は「コスト最小化」や「距離最短化」といった単純な数値目的ではなく、**異なるスケジュールアルゴリズム間の関係性を安定に保つこと**を指向していると解釈できる。言い換えれば、AI エージェントは解の質そのものよりも、**解が生成され続ける枠組みの持続可能性**を評価関数として持つ。

この観点に立つと、AI エージェントの学習や調整は、個々のスケジュール解の成功・失敗ではなく、「どのアルゴリズム構成がどの状況で破綻しにくい」という構造的知識の蓄積として理解される。これは、AI を最適化器としてではなく、**問題構造の管理者**として用いる本レポートの立場と整合的である。

## 7 まとめ

本レポートでは、スケジュール問題における AI 活用を、アルゴリズムの高度化ではなく、**アルゴリズム間の関係設計**として捉えた。既存の数理最適化アルゴリズムは依然として強力であるが、現実の複雑さを直接内部に取り込むことには限界がある。AI エージェントを計算論的インタフェースとして配置し、複数のアルゴリズムを状況に応じて組み合わせることで、説明可能性と実用性を両立したスケジュール設計が可能になると考えられる。

## A Mathematica による数式モデル生成・検証の役割

本付録では、本文で議論した (1) AI エージェントによる数式モデル・制約表現の選択、(2) 数式処理システムによるモデル生成・検証、(3) 既存スケジュールアルゴリズムによる解決、という三段構成のうち、特に (2) における Mathematica (Wolfram Language) の役割を、物流スケジューリングを前提として具体的に整理する。

## A.1 物流スケジュールにおける数式表現の難しさ

物流分野のスケジュール問題は、配送経路や積載制約といった離散的構造を持つ一方で、以下のような連続量・条件依存性を強く含む。

- 積載量・重量・体積に関する連続制約
- 配送時間帯制約と交通状況による到着時刻の揺らぎ
- 倉庫内作業時間や積み下ろし時間の非線形性
- 燃料消費量や CO<sub>2</sub> 排出量と走行条件の関係

これらは、線形計画法や最短経路問題に直接与えられる「固定された制約」や「静的な重み」として記述することが難しく、通常は強い近似や単純化が施される。その結果、現実との乖離が生じ、頻繁な再調整や例外処理が必要となる。

## A.2 AI エージェントと Mathematica の役割分担

本文で述べた立場では、AI エージェントは物流スケジュールを直接生成する主体ではなく、現実状況を観測した上で「どの数式モデル・制約表現を用いるか」を選択する役割を担う。このとき Mathematica は、その選択結果を受け取り、以下の機能を提供する。

- 現実条件を反映した数式モデルの生成
- 制約条件の変形・簡約・近似
- 生成されたモデルの整合性・妥当性検証

すなわち、Mathematica は「現実を直接スケジュールアルゴリズムに押し込む」のではなく、「既存アルゴリズムが扱える形式へ翻訳・検証する中間層」として機能する。

## A.3 物流における具体的活用場面

### A.3.1 条件依存制約の生成

物流では、積載量や配送順序に応じて制約条件自体が変化する場合が多い。例えば、車両の積載率が高いほど走行速度が低下する、あるいは積み下ろし作業時間が非線形に増加する、といった関係である。

Mathematica はこれらの関係を記号的に保持し、AI エージェントが選択した前提条件（近似の粒度、無視してよい要因など）に応じて、線形制約、区分線形制約、あるいは条件付き制約として自動的に生成する役割を果たす。

### A.3.2 時間スケールの切り替え

物流スケジュールでは、日次計画・時間帯計画・リアルタイム調整といった異なる時間スケールが混在する。Mathematica は、同一の数式モデルを時間分解能の異なる形で再構成し、それぞれに適した制約集合を導出することができる。

これにより、AI エージェントは「詳細なモデルを使うべき状況」と「粗いモデルで十分な状況」を切り替え、既存のスケジューリングアルゴリズムを使い分けることが可能となる。

### A.3.3 スケジュール解の数理検証

既存アルゴリズムが生成したスケジュールに対し、Mathematica はその解を関数・写像として扱い、制約違反が生じる条件範囲やパラメータ依存性を解析できる。これは、「この解がどの条件で破綻するか」を事前に把握するための検証層として機能する。

この検証結果は AI エージェントにフィードバックされ、次回以降のモデル選択やアルゴリズム組み合わせに反映される。

## A.4 位置づけの整理

以上より、Mathematica は物流スケジューリングにおいて、最適解を直接計算するためのツールではなく、以下の役割を担う。

- 現実条件を数式として表現・操作するための基盤
- 制約や目的関数の構造を可視化・検証する手段
- AI エージェントと既存スケジューリングアルゴリズムを接続する計算論的インタフェース

この位置づけにより、物流スケジュールの複雑さは単一アルゴリズムの内部に押し込められるのではなく、AI エージェントによるモデル選択と、Mathematica による数式生成・検証を介して段階的に吸収される。これは、本文で述べた「アルゴリズムの組み合わせによって現実の複雑さに対応する」という設計思想を、数式処理レベルで具体化するものである。