

誘導加熱焼入れにおける シミュレーションによる最適条件導出方法の検証

岸 智 裕

1 はじめに

製造業において、製品品質の安定化、生産性の向上、コスト削減を同時に達成するためには、各種製造条件の最適化が不可欠である。従来、製造条件の設定は、熟練者の経験や試行錯誤に基づく実験的手法に依存しており、条件出しに多大な時間と工数を要することが課題となっていた。

近年、数値シミュレーション技術の進展により、製造プロセスを仮想的に再現し、条件変更に対する出力の変化を定量的に評価することが可能となっている。これにより、実験回数を大幅に削減しつつ、目的に応じた最適条件を効率的に導出するアプローチが現実的な選択肢となりつつある。

本報では、製造条件の最適化におけるシミュレーション活用の有効性を、誘導加熱焼入れ条件の最適化を題材として検証した事例について述べる。対象としたのは、建設機械用ピストンロッドの段付き形状部分に対する誘導加熱焼入れである。対象の工程は、ねじ部外径などの加工が施された複雑な形状のワークに対し、定置加熱と移動加熱を組み合わせたプロセスで、一度に複数部位へ熱処理を施している。そのため、加熱条件の設計変数が多岐にわたり、従来の実験的手法では最適条件の導出が困難であるとともに、品質確認を含む工数が膨大になるという課題があった。これらの課題に対し、電磁界解析ソフトウェアJMAG[®]（以下、JMAG[®]）を用いた誘導加熱解析における最適条件探索を、設計探索ツールSimcenter HEEDS[®]（以下、HEEDS[®]）あるいはJMAG[®]内蔵の遺伝的アルゴリズムを用いて検証し、それぞれのアプローチによる最適条件導出の過程と結果を比較・考察する。さらに、実業務でのシミュレーションによる最適化計算活用の検証について述べる。

2 対象工程の概要

図1に対象工程の概略図を示す。対象工程では、予めねじ部外径などの旋削加工が施された段付き丸棒に焼入れ、焼戻しを施すため、以下の3段階の加熱操作が組み合わされている。

- ①細径部に対する定置加熱（以下、予熱）
- ②太径部に対する定置加熱（以下、定置加熱）
- ③細径部方向への移動加熱（以下、移動加熱）

このような複雑な加熱工程により、異なる外径部位に対して一度に熱処理を施しているが、加熱条件の設計変数が多岐にわたるため、従来の実験的手法による最適条件探索は極めて困難であった。具体的には、図1に示すような予熱、定置加熱、移動加熱に関する9つの設計変数が存在し、これらの組み合わせにより温度分布や加熱時間が大きく変動する。また、熱処理品質を確保するためには、各部位の加熱温度が所定の範囲に収まるよう制御する必要がある。対象工程では、生産性向上のため熱処理工程のCT（Cycle Time）の短縮が求められており、現状熱処理設備で実現可能な最速条件を導出するため、大幅な条件変更の必要があった。

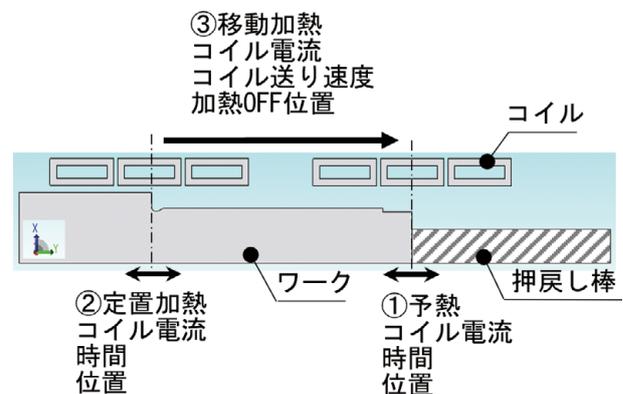


図1 対象工程の概略図

3 焼入れの基礎

焼入れは、鋼材の機械的性質を制御するための代表的な熱処理工程であり、鋼をオーステナイト化温度まで加熱した後、急冷することでマルテンサイト組織を形成し、硬さおよび強度を向上させる処理である。

焼入れにおいて、加熱不足は硬化不良を招く。一方で、過度な加熱も結晶粒粗大化や酸化、さらには焼割れの原因となるため、焼入れ部位の加熱温度は適切な温度範囲に制御する必要がある。

一般的に、焼入れされた鋼は硬い反面靱性を失うため、再度適切な温度に加熱する焼戻しを行うことで、靱性の回復や残留応力を緩和させる。本検証では、焼入れのみを扱う。

4 検証用解析モデルの概要

図2に本検証で用いたモデルを示す。本検証では、最適化計算の動作検証に重点を置き、単一ケースの計算を可能な限り軽くするため、対象工程のワーク及び加熱コイルを簡略化し、半径 r 部と $2r$ 部を持つ段付き丸棒の10deg軸対称分割モデルとした。ワーク表面には4点の温度測定点（プローブ）を設定し、各点の最高温度を出力する。図3にプローブ位置を示す。ワーク材料データには、材料物性値計算ソフトウェアJMatPro[®]で計算したS45Cの温度依存性データを使用し、加熱コイルにはJMAG[®]内蔵の銅のデータを使用した。なお、加熱コイルは内部の水路によって水冷されており、温度依存性の影響

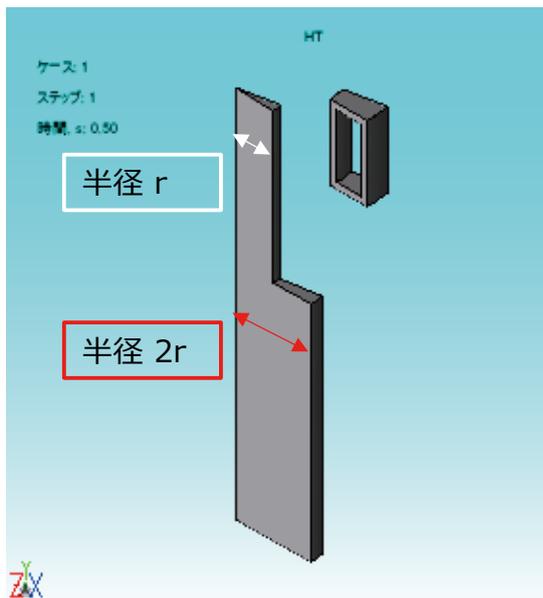


図2 本検証で用いたモデル

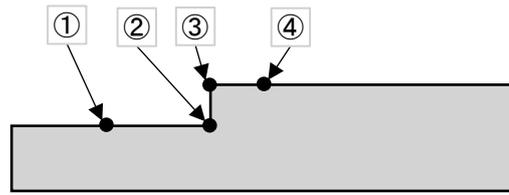


図3 プローブ位置

は小さいと考え、モデル簡略化のため加熱条件を入れず、初期温度20℃のままとした。

メッシュサイズやタイムステップについても、計算を可能な限り軽くするため妥当な値で一定とし、JMAG[®]の周波数応答断面磁界解析と熱解析の連成解析が実施されるよう設定した。

設計変数は図1に示した9個の設計変数を設定した。表1に設計変数及び探索範囲を示す。本検証では、「焼入れ範囲において均一な焼入れが可能で、かつ処理時間が最も短い条件を見出す」というユースケースを想定し、目的関数は図3に示したプローブ各点の最高温度の標準偏差及び加熱時間の最小化とする多目的最適化問題とした。ここで、加熱時間は予熱開始から移動加熱終了までの時間を示す。制約条件は、プローブ各点の温度及びワーク全体での最高温度に設定した。表2、3に目的関数及び制約条件をそれぞれ示す。

表1 設計変数及び探索範囲

	設計変数	Min	Max
予熱	位置 [mm]	-10	10
	時間 [s]	0	20
	出力 [V]	10	60
定置加熱	位置 [mm]	-60	-40
	時間 [s]	0	10
	出力 [V]	10	40
移動加熱	出力 [V]	10	60
	加熱OFF位置 [mm]	-10	10
	コイル送り速度 [m/s]	0.001	0.01

表2 目的関数

項目	方向
プローブ各点の最高温度標準偏差	最小化
加熱時間	最小化

表3 制約条件

項目	制約条件
プローブ各点の最高温度 [°C]	≥850
ワーク全体の最高温度 [°C]	≤1100

5 最適化手法の概要

5.1 探索戦略の分類

最適化アルゴリズムにおける探索戦略は、大きく「局所探索 (local search)」と「大域探索 (global search)」に分類される。

局所探索は、現在の解の近傍を重点的に探索する手法であり、収束速度が速く、微細な調整に適している。一方で、探索範囲が限定されるため、局所最適解に陥るリスクがある。

大域探索は、探索空間全体を広く探索することで、より良い解を見つける可能性が高いが、計算コストが大きく、収束までに時間を要する場合がある。

5.2 SHERPAアルゴリズム¹⁾

SHERPA (Simultaneously Hybrid Exploration that is Robust, Progressive and Adaptive) は、HEEDS[®]に搭載された独自の最適化アルゴリズムであり、局所探索と大域探索の複数の手法を同時並行的に活用するハイブリッド型の適応的探索戦略を特徴としている。このアルゴリズムは、2～10種類の探索手法を並列に実行し、それぞれの手法の有効性に応じて使用比率を動的に調整することで、探索効率を高めている。また、各探索手法に含まれる内部パラメータは、探索中に得られる設計空間の情報をもとに自動的に調整されるため、ユーザによる事前のチューニングは不要である。さらに、探索の過程で設計空間の構造を学習し、探索戦略を柔軟に変更することで、複雑な設計問題にも対応可能となっている。

これらの特徴により、SHERPAは他の代表的な最適化手法と比較して、同じ評価回数でより高品質な解を導出する傾向があり、初期条件をランダムに変えて複数回試行した場合でも、標準偏差が小さく安定した性能を示すなど、頑健性にも優れている。これらの利点により、ユーザは探索手法に関する専門知識がなくても、効率的に最適化問題を実行することが可能となっている。

5.3 遺伝的アルゴリズム

遺伝的アルゴリズムは、大域探索法の一つで、生物の進化過程における「自然淘汰」「交叉」「突然変異」などのメカニズムを模倣した探索手法であり、確率的な最適化アルゴリズムの一種である。探索空間における複数の解候補 (個体) を集団として扱い、世代を重ねながら最適解に近づけていく。基本的な処理手順は以下の通りである²⁾。

- ①初期集団の生成：設計変数の範囲内でランダムに個体を生成
- ②適応度評価：各個体に対して目的関数や制約条件に基づく評価値 (適応度) を算出

- ③選択 (Selection)：適応度に応じて次世代に残す親個体を選定
- ④交叉 (Crossover)：親個体の設計変数を組み合わせて子個体を生成
- ⑤突然変異 (Mutation)：子個体の一部設計変数をランダムに変更
- ⑥世代更新：新たな集団を構成し、再び評価・選択を繰り返す

このような進化的操作を繰り返すことで、探索空間全体を広く探索しながら、局所最適に陥ることなくグローバルな最適解を導出することが可能となる。

JMAG[®]では、多目的最適化手法としてNSGA-II (Non-dominated Sorting Genetic Algorithm II) が実装されており、以下の特徴を持つ³⁾⁴⁾。

- ①高速な非優越ソートによる個体のランク付け
 - ②多様性保持のための混雑度評価
 - ③エリート主義による優良個体の保存
- これにより、複数の目的関数間のトレードオフ関係を考慮したパレート最適解^{注1)}の集合 (パレートフロント^{注2)}) を効率的に探索することが可能となっている。

注1) 他のどの解にも優越されないような解⁵⁾。

注2) パレート最適解が形成する面⁵⁾。

6 最適化計算の設定と実行環境

6.1 HEEDS[®]による最適化計算設定

HEEDS[®]での最適化計算の設定は、基本的には表1～3の設計変数、目的関数及び制約条件、そして計算回数である。表4に各試行の計算回数を示す。表中の100+100といった表記は、100回の計算完了後に100回追加計算を実行したことを意味する。アーカイブサイズとは、一度に作成されるケース数であり、作成されたアーカイブサイズ分の計算結果から次のアーカイブサイズ分のケースを生成する、という動作を指定計算回数に達するまで繰り返す。この一連の動作をサイクルと呼ぶ。本検証では、アーカイブサイズはデフォルト値のまま計算を実行した。

表4 各試行の計算回数

試行	計算回数	アーカイブサイズ
1	100+100	20 (デフォルト値)
2	200+100	
3	200	
4 ^{注3)}	5000 (424)	

注3) 試行4は計算回数設定を5000回としたが、諸般の都合により424回で計算を中断した。

6.2 JMAG[®]による最適化計算設定

JMAG[®]での最適化計算の設定は、表1～3の設計変数、目的関数及び制約条件に加え、初期ケース生成方法、世代数、集団サイズを指定する。表5にJMAG[®]最適化各試行の設定を示す。初期ケース生成方法の「既存ケースの使用」では、解析スタディを設定した際に生成される1ケースを除き、集団サイズ分のケースに対し各設計変数が指定した探索範囲内でランダムに割り当てられる。また、集団サイズ、世代数はJMAG[®]の推奨値とされている、設計変数×10を参考に設定し、試行3では集団サイズの影響を把握するため、試行1及び2の半分の値を設定した。

表5 JMAG[®]最適化各試行の設定

試行	集団サイズ	世代数	初期ケース
1	96	100	既存ケースの使用
2	96		
3	48		

6.3 実行環境

表6に各最適化計算の実行環境を示す。HEEDS[®]最適化は通常のローカルPCで実施した。JMAG[®]最適化はケース数が膨大になるためPSL^{注4)}の使用及びAmazon Web Services[®]のクラウドコンピューティングサービスであるAmazon Elastic Compute Cloud (EC2)を利用した。本検証では、計算用インスタンスの物理コア数は、表5の集団サイズ数に合わせ、1世代分の計算を並列実行する設定とした。

7 計算結果及び考察

7.1 HEEDS[®]による最適化計算

図4にHEEDS[®]による最適化計算で得られた実行可能解^{注5)}分布を示す。試行1では、最初の100回の

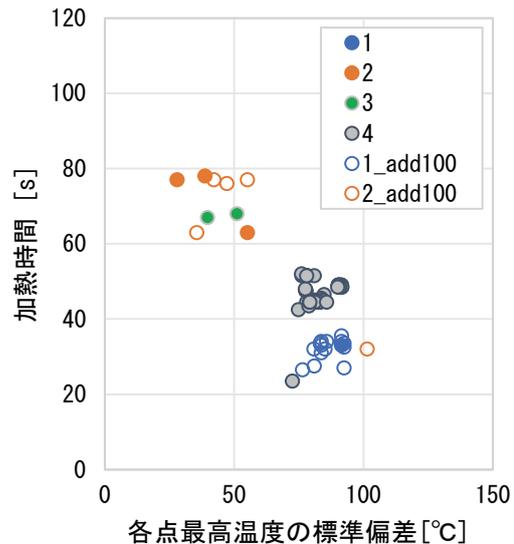


図4 HEEDS[®]による最適化計算で得られた実行可能解分布

注5) すべての制約条件を満たした解。

計算で1通りの実行可能解が得られ、追加の100回の計算ではその周辺に実行可能解が集中した(1_add100)。一方、計算回数を200回で指定した試行2及び3では、試行1とは異なる領域の実行可能解が得られ、試行2における100回の追加計算では、最初の200回の計算で得られた実行可能解の周辺に加え、試行1の実行可能解の周辺領域の解が得られた(2_add100)。また、試行4でも試行1、試行2、3とは異なる領域の実行可能解が得られた。これは、第5章で述べたように、SHERPAが自動で複数の探索手法を適用し、設計空間に合わせて各種パラメータを自動チューニングすることに起因していると考えられる。また、試行2における100回追加計算時(2_add100)の実行可能解の得られ方から、HEEDS[®]の特徴である複数の局所探索及び大域探索アルゴリズムを同時に使い分けながら探索が進んでいる様子がうかがえる。

表6 各最適化計算の実行環境

手法	ソフトウェア	解析環境	備考
HEEDS [®] 最適化	JMAG v22.1 HEEDS 2304	ローカルPC	CPU : Intel [®] Xeon [®] W-2133 CPU @ 3.60GHz RAM : 32.0GB 物理コア数 : 6 OS : Windows 10
	JMAG v23.1 HEEDS 2404 (試行4のみ)		
JMAG [®] 最適化	JMAG v23.1	EC2 計算ノード : c6a.metal, c6a.24xlarge	PSL使用 CPU : AMD EPYC 7R13 RAM : 384GB (.metal) 192GB (.24xlarge) 物理コア数 : 96 (.metal) 48 (.24xlarge) OS : Amazon Linux 2

注4) PSL (Power Simulation License) : 100ケースまで並列実行可能なJMAGオプションライセンス。

表7にHEEDS[®]各試行の計算時間を示す。本検証では、HEEDS[®]による最適化計算では並列実行をしていないため、生成されたケースを順次計算している。計算時間はモデルの要素数やステップ数に依存するが、本検証において要素数が2次元磁界解析で約6000、3次元熱解析で約30000、ステップ数は180以下の範囲で探索した場合、300ケース程度の計算であれば1週間程度で完了する。

表7 HEEDS[®]各試行の計算時間

試行	計算回数	計算時間
1	100+100	約72h
2	200+100	約157h
3	200	約92h
4	424	約196h

7.2 JMAG[®]による最適化計算

図5にJMAG[®]による最適化計算で得た実行可能解分布を示す。JMAG[®]による最適化では、広い領域にわたる実行可能解が得られ、試行3 > 試行2 > 試行1の順でその範囲が広く見える。各試行の実行可能解分布の差は、初期ケースの生成方法に起因すると考えられる。すなわち、ランダムに生成された初期集団が出力した解によって探索領域が変化することが考えられる。

試行3では集団サイズを減少させた影響がみられた。図6に試行3の実行可能解分布を示す。試行3では赤丸で図示した領域の解が極端に少ないことがわかる。遺伝的アルゴリズムにおいて、集団サイズを縮小することは探索空間を縮小することにつながり、探索漏れによって本来実行可能解であるはずの設計変数の組み合わせに辿り着くことができなかつたためと考えられる。また、前述の通り、初期ケースの解の分布によって最終的な実行可能解分布が変わる可能性を考慮すると、探索漏れが発生する領域はランダムであると考えられるため、集団サイズはむやみに縮小するべきではない。

表8にJMAG[®]各試行の計算時間を示す。JMAG[®]最適化計算では、PSL及び並列実行数分の物理コアを持ったEC2インスタンスの使用によって、1世代分の計算はおおよそ1ケース分の計算時間で完了するため、本検証の解析データサイズであれば、集団サイズ96、世代数100という大規模な計算も約2日で完了する。

7.3 JMAG[®]最適化計算のパレートフロントとHEEDS[®]実行可能解分布

図5からもわかるように、JMAG[®]最適化計算に

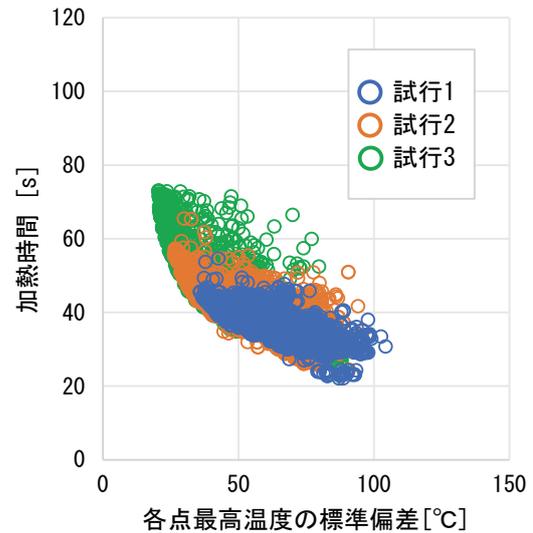


図5 JMAG[®]による最適化計算で得られた実行可能解分布

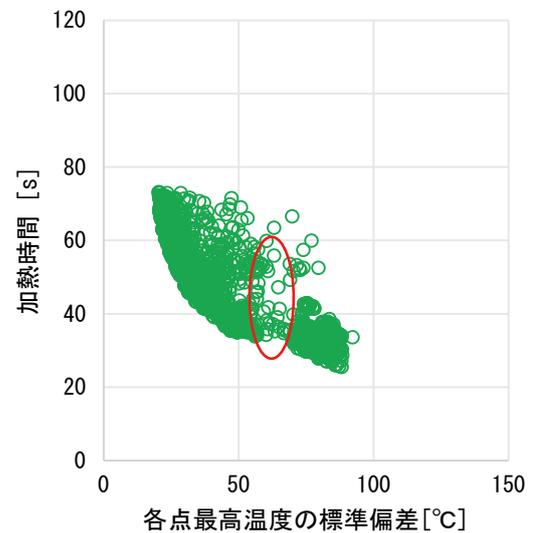


図6 試行3の実行可能解分布

表8 JMAG[®]各試行の計算時間

試行	計算回数	計算時間
1	9696	約45h
2	9696	約49.5h
3	4848	約35.5h

よる実行可能解分布はパレートフロントを形成している。図7にJMAG[®]パレートフロントとHEEDS[®]実行可能解分布を示す。なお、パレートフロントは実行可能解の範囲で、各点の最高温度の標準偏差の最小値から最大値までを1刻みで探索し、探索幅において加熱時間が最小の値を抽出した。

JMAG[®]の試行1, 2は同等のパレートフロントを形成し、試行3では探索漏れが発生した領域におい

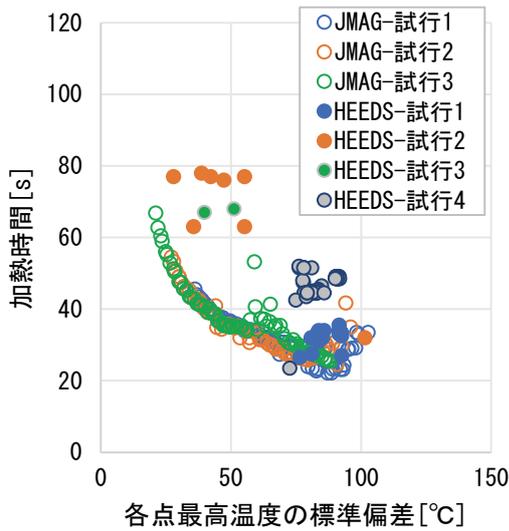


図7 JMAG[®]パレートフロントとHEEDS[®]実行可能解分布

て不連続となったが、加熱時間が長時間側のパレートフロントは同等の領域だった。パレートフロントが形成されることで、設計者は二つの目的関数間のトレードオフを把握し、任意のバランスで限界設計が可能となる。

一方で、HEEDS[®]が出力した実行可能解分布は、JMAG[®]が導出したパレートフロントに対して全体的に劣解ではあるものの、HEEDS[®]の試行1及び試行4の実行可能解はJMAG[®]で導出されたパレートフロントと同等もしくは超えた解（パレートフロントより左下の領域の解）を導出している。また、HEEDS[®]の試行4で導出された、JMAG[®]で導出されたパレートフロントを超えた解は、HEEDS[®]試行4の解集合から離れた領域である。これは、7.1節でも言及したように、SHERPAアルゴリズムが局所探索と大域探索を組み合わせることで効率的に設計空間を探索していることを示していると考えられる。このことにより、HEEDS[®]は一般的な遺伝的アルゴリズムによる探索と比較してかなり少ない計算回数で大域最適解やパレートフロントを導出し得ると考えられる。

8 実業務での最適化計算活用の検証

第7章までの検証にて、最適化計算活用による熱処理条件導出の有効性を示し、対象工程の実業務での検証を実施した。本検証では、最適化計算手法は、少ない計算回数で効率的に設計空間を探索したSHERPAを選択した。また、焼入れ範囲において最高温度が狙いの温度範囲にあれば品質を担保できるため、前述の検証で目的関数としていた各点の最

高温度の標準偏差を除外し、加熱時間を最小化する単目的最適化問題とした。また、解析モデルや探索範囲、制約条件についても実機モデルに合わせて修正した。

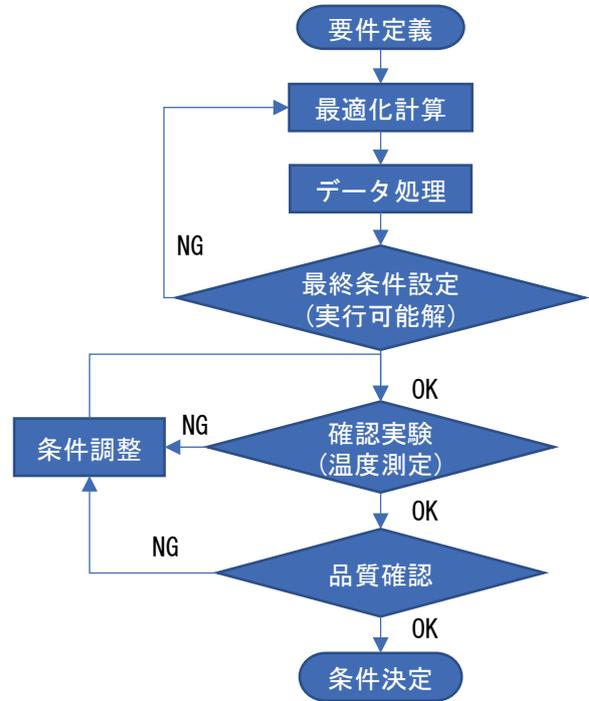


図8 改善条件決定フロー

図8に改善条件決定フローを示す。大まかな順序としては以下の通りである。

- ①対象工程の目標サイクルタイムや制約条件を踏まえた探索範囲、目的関数、制約条件を設定し、与えられた期間で計算可能な回数を設定し最適化計算を実施する。
- ②最適化計算が完了したらポスト処理を実施し、実行可能解を抽出し条件を設定する。
- ③設定した条件にて実際に量産設備で焼入れを実施し、解析モデルと同等の箇所まで表面温度を測定し、狙い温度範囲に入っているか確認する。
- ④品質確認で問題なければ条件を決定する。

本検証では、途中でモデル修正などを加えながら最適化計算を実施し、計610回の計算を実施した。その内、最終モデルでの264回の計算の中で得た6個の実行可能解の中からベスト条件を選択して温度測定実験を実施した。

図9に解析結果、実測値及び条件調整後実測値を示す。純粋に最適化計算結果から設定した条件で焼入れすると、実測値は解析結果に対して高めの温度になった。また、地点1では狙いの温度範囲を超える最高温度となった。これはJMAG[®]解析モデルの合わせこみが不十分で、物理現象を再現できていな

いたためと考えられ、今後改善が必要である。本検証では、この結果から3回ほど熱処理条件を微調整することで、各部が狙いの温度範囲内となる熱処理条件を導出することができ、従来の量産条件と比較してMCT (Machine Cycle Time) は29%短縮、電力消費によるCO₂排出量も24%削減した。MCTの中でも、最適化計算の対象である加熱に要する時間は49%削減された。

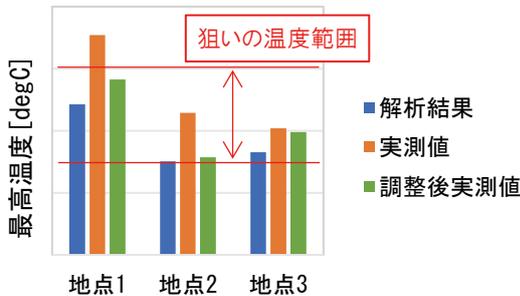


図9 解析結果、実測値及び条件調整後実測値

ここで、同様の改善を実験での試行錯誤によって実施した場合と比較する。本検証を実施する以前に、担当者が改善条件の導出に実際に費やした工数を概算し、シミュレーションによる最適化計算を活用した場合と比較した。図10に概算工数の比較を示す。結果的に、最適化計算の活用によって、約76%の工数削減を実現した。

実験による試行錯誤では、対象工程の複雑な加熱工程によって、予熱、定置加熱、移動加熱それぞれの傾向把握を別々に実施した結果から条件を組み合わせる必要がある。よって、多くの実験や品質確認が必要であることに加え、担当者が保有しているノウハウに強く依存する。さらに、実際に実験可能な水準は工数や納期の面で限定的であり、導出した条件が「最適であるか」に関しては疑念が残る。

一方で、最適化計算を活用する場合、実験、品質確認工数を大幅に削減することができる上、数百通りの試行結果を確認することが可能で、担当者が保有しているノウハウに関係なく最適条件を設定可能

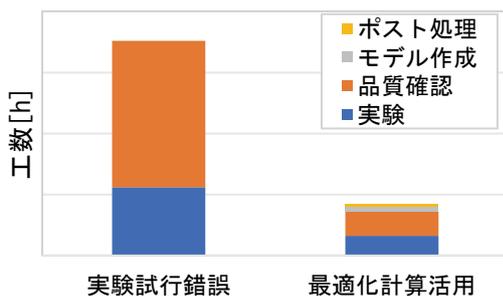


図10 概算工数の比較

となる。さらに、最適化計算の実行中は他の業務に従事することができ、業務効率向上が可能となる。また、本検証では並列実行せずに順次計算を実施したが、人の工数削減による費用効果を計算リソースに割いて並列計算を実行すれば、最適化計算に要する期間のさらなる短縮、あるいは計算回数の増加によるさらなる設計探査が可能である。

本検証結果から、解析結果と実験結果の乖離はあるものの、熱処理条件の目途付けには活用可能であり、最適化計算活用の一定の効果を確認できた。

9 おわりに

建設機械用ピストンロッドの段付き形状部分への高周波焼入れ工程を対象に、シミュレーションによる条件最適化の活用検証を実施し、その有効性を確認した。生産技術研究所では、過去にもめっきのマスクング治具の形状最適化をHEEDS®を活用して実施した実績もあり⁶⁾、最適化計算が普及しつつある。

第8章では、解析結果と実験結果の乖離があり、確認実験において条件の微調整を必要とした。今後、解析精度の向上に取り組み、解析結果の信頼性を向上させていく。

また、最適化計算は加工技術だけでなく、製品設計やプラントシミュレーションなど、あらゆる設計探査への活用が可能である。今後、デジタルツインの実現に向けて、様々な分野のシミュレーション技術を確認し、最適化計算を適用することで、劇的な生産性および業務効率向上を実現していく。

最後に、本検証においてご協力いただいたソフトウェアベンダー様、並びに社内関係部門の方々にご場をお借りして熱く御礼を申し上げる。

参考文献

- 1) Red Cedar Technology: SHERPA - An Efficient and Robust Optimization/Search Algorithm, <https://www.redcedartech.com/pdfs/SHERPA.pdf>
- 2) 北野宏明：遺伝的アルゴリズム，人工知能学会誌，Vol. 7, No. 1, pp. 26-37, (1992年1月)
- 3) K. Deb, A. Pratap, S. Agarwal, T. Meyarivan: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II, IEEE Trans. On Evolutionary Computation, Vol. 6, No. 2, pp. 182-197 (2002/4)
- 4) 株式会社JSOL: JMAG最適化マニュアル, (2024年12月)
- 5) 渡邊真也：パレート解析のための技術動向，システム/制御/情報，第60巻，第7号，pp. 272-277, (2016年7月)
- 6) 水野祐樹，寺戸彩由子：めっきシミュレーションによ

る膜厚分布均一化, カヤバ技報, 第65号, pp. 3-7, (2022年10月)

著 者



岸 智裕

2018年入社. 技術本部生産技術研究所第一研究室. 熱処理技術の開発に従事.