

CFDシミュレータを用いた最適化手法に関する研究

2011年2月

博士（工学）

学位論文：博士(工学) 甲399号

栗山嘉文

岐阜大学

CFD シミュレータを用いた最適化手法に関する研究

栗山 嘉文

生産開発システム工学専攻

要旨

CFD シミュレータは、近年のコンピュータの演算能力の向上と低価格化により、様々な分野で設計・解析支援ツールとして広く利用されるようになり、一部の分野ではこのシミュレータを援用して性能向上を目的とした最適化計算（以下：CFD 最適化問題）へ利用するまでに至っている。しかし CFD 最適化問題には解探索に膨大な計算時間が必要な上に、CFD シミュレータから得られる数値が形成する解空間は不安定であるため、多数の局所解が分布する多峰性解空間における最適化問題を解く必要がある。そこで本研究では CFD シミュレータを援用した最適化の設計・構築法を提案するとともに、少ない個体数でも良好な結果を得られる最適化アルゴリズムを構築する。

CFD シミュレータを援用した最適化として 6 自由度マニピュレータを用いた溢流抑制制御を行った。具体的には、スプーンに入った液体の挙動をシミュレータと一致させ、スプーンいっぱいに満たされた液体を溢流させない搬送制御入力 of 導出を行った。結果、高速かつ溢流を抑制させた搬送を行うことが可能となった。ダイカストプランジャの最適速度入力の導出においては、その欠陥原因を流体解析シミュレータを用いて解析を行い、欠陥の原因の一つである空気巻き込み量を抑制するプランジャの最適速度入力を求めた。結果、実験を通して空気巻き込みの欠陥を低減させる最適速度入力を求めることが可能となった。一方で、これらの最適化においては多峰性解空間に対して有効である遺伝的アルゴリズム最適化手法を適用していた。

しかしながら CFD シミュレータを援用した場合、一度の解析に数分～数時間程度の計算時間が必要であったため、最適化を行う際には必然的にその個体数 (探索点) を低減する必要があった。それに対して遺伝的アルゴリズムは多くの個体数を用いて解析することができるという前提で設計されているため、少ない個体数で解を見つける場合は非常に局所解に陥り易く、最適解を見つけるまでに幾度かの試行を行う必要があった。特に、解を算出しても良好な結果を得ることができないという問題点があった。そこで、実験でも良好な結果となる解は大域的に良好な評価を得ているという考えに基づき、実験でも良好な結果を得られるような解の算出手法として極値近傍探索アルゴリズムを構築した。具体的には得られた評価値から近似曲線を生成し、その近似曲線が形成した極値を判別することで大域的に良好な評価を得ている空間を予測できると考えた。ダイカストにおける射出速度最適化問題に極値近傍探索アルゴリズムを適用した結果、実験でも良好な結果を得ることが可能となった。

さらなる探索性能の向上をするため、これまでの個体生成および配置を確率的手法から制御則を応用した手法へと変える。CFD 最適化問題に対して収束性能に大きな差

を生む要因として探索点 (個体) の配置があげられる。研究ではその探索点の配置をメルセンヌツイスター法の乱数を利用したランダムによる配置を行っていた。しかしながら、個体数が少ない場合で配置を行った場合、探索点が密もしくは疎となる箇所が発生し、結果的に探索領域に探索点配置に偏りを生じさせてしまうことで、未解析領域ができやすくなる。その結果、局所解へ非常に陥りやすい状況を作ることとなる。一方で、良好な解空間を集中的に探索することが収束性能の向上に必要不可欠であることがわかった。そこで、探索点の偏りを調節し未解析領域を抑制しつつ、良好な解空間では探索点を集中させて探索することができる最適化アルゴリズム (多極集中型最適化アルゴリズム) を提案した。具体的には、探索点を偏り無く、解析領域全体へ配置させるための分散制御則として、被覆制御の考え方を応用し、探索点自身が移動して他の個体との間隔の調整を行うようにする。空間全体への探索点の分散後、探索点周辺に既に評価されている点がある場合、その評価を参照して、探索点同士の間隔を保ちつつ移動する法則を探索点に与える。これにより解の向上が見込まれる空間へと探索点を集中させることができ、収束性能の向上や未解析領域も抑制することが期待できる。

また実験でも良好な結果を得るためには、シミュレータの計算誤差などの影響を考慮し、大域的に良好な評価を得ている解を見つける必要があった。しかしながら極値近傍探索アルゴリズムなど近似曲線を利用して探索する場合、設定する近似曲線の次数に大きく左右されてしまう。そこで大域的に良好な解を導出するアルゴリズムを構築することを目的として、クラスター分析を応用した探索点の密度で良好な空間を判断することのできるアルゴリズムを提案した。

本研究で提案した最適化アルゴリズムを用いることで実験でも良好な評価を得ることのできる解を効率良く求めることができると共に、ダイカストにおけるプランジャチップ形状の最適化や、ダイカスト以外の、例えば船舶の形状最適化など様々な設計最適化支援ツールとして用いられることが期待でき、生産準備段階におけるコストの削減や、設計期間の短縮などの効果が期待できる。

A study on optimization methods using CFD simulator

Yoshifumi Kuriyama

Mechanical and Civil Engineering Division, Gifu University

ABSTRACT

In recent years, numerical simulators for fluid analysis based on computational fluid dynamics (CFD) have been used to study the behavior of fluid around an object and its thermal hydraulics. CFD is a technique based on the Navier-Stokes equations, as well as on energy and mass conservation laws. Recently, with the development of computing power and the price plummet of personal computers, the optimization algorithm has been applied an CFD simulator to finding the optimal control settings or optimal shape.

However, in cases where the problem has many local solutions due to the effect of the loss of significant digits or underflow error, the most widely used general optimization algorithm like Genetic Algorithm (GA) is highly likely to find a local solution. Of course, this problem can be solved by enlarging the number of population members, the number of generations and the mutation evolution; on the other hand, the computational time for one condition lasts only a few minutes and the optimization requires hundreds of repeated computations. Thus the optimization using the CFD simulator needs a lot of time to finish the task. In addition, a simulation with low computational cost does not always correspond to actual production of a high-quality product because of the unstable solution space.

In this paper, as the optimization with CFD simulator, two optimization method are introduced. In the transferred control considering spilling avoidance, in the near future, there will be a significant problem obtaining workers in the fields of welfare and nursing care because of labor shortage. To solve this problem, many welfare robots, such as an upper extremity motion assistance robot and a meal assistance robot, have been studied. Patients usually drink soup or beverages at meals, but meal assistance robots with liquid transfer control have not yet been developed. Thus, a 3-D transfer control system for meal assist robots that incorporates spilling avoidance is designed.

Another method is calculation of the optimal plunger velocity input for die-casting. In this background, die casting is adequate for mass production and it has the advantage of being able to yield a complicated shape accurately. A defect of die casting for high-speed injection molding is the occurrence of shrinkage cavities. To avoid the shrinkage cavities, velocity control of a plunger is very effective. However, the velocity control input is manually constructed by skillful workers because analysis behavior of molten metal is very difficult and the injection molding is performed at invisible places in the mold. Thus optimum veloc-

ity control input of die casting plunger is calculated in order to enable the reduction of air entrainment and prevent the occurrence of shrinkage cavities in die casting product.

Another aim of this study is to design a solution search algorithm that requires a smaller computational time to find a solution which can be applied to the production of high-quality products. In this study, two optimization algorithms are proposed, and the proposed methods apply the optimal plunger velocity problem in die-casting product. One method is the Extremal Distribution Sorting Algorithm (EDSA), which searches intensively by using approximated curve and evaluated value at the improved point in the nearly extremal solution. Another method is Multi-subcenters Solution Search Algorithm(MSSA), which consists of a distribution control to handle the distribution of search points to expand the clearance between search points, and the convergence control to control the density of search points to converge the predicted high quality solution space. Furthermore, to create the performance of searching the point to produce the high-quality products for the MSSA, Space Searching Algorithm(SSA) is designed. The effectiveness of the proposed systems are shown through simulation and experimental results.

目次

第1章 緒言	1
1.1 研究背景	1
1.2 従来研究	3
1.2.1 CFD シミュレータを用いた研究	3
1.2.2 最適化アルゴリズムの研究	4
1.2.3 CFD シミュレータを援用した最適化の研究	5
1.3 CFD シミュレータを援用した最適化における問題点	6
1.4 研究目的と構成	9
第2章 6自由度マニピュレータを用いた溢流抑制制御	11
2.1 概要	11
2.2 実験装置	12
2.3 CFD シミュレータの設定	13
2.4 モデリング	17
2.4.1 運動学	17
2.4.2 逆運動学	19
2.4.3 モータモデルの同定	24
2.5 溢流の防止を考慮した搬送制御系設計	35
2.5.1 食事支援ロボットによる液状食物の搬送	35
2.5.2 加速カーブの定式化	36
2.5.3 遺伝的アルゴリズム	38
2.5.4 最適加速カーブの導出	40
2.6 液体ハンドリング制御実験	42
2.6.1 軌道設計	42
2.6.2 3次元搬送	43
2.7 まとめ	49
第3章 ダイカストプランジャの最適射出速度入力の導出	51
3.1 概要	51
3.2 ダイカストにおける射出速度の最適化	52
3.2.1 実験装置	52
3.2.2 欠陥の解析	54
3.2.3 評価関数の構築	63

3.3	シミュレーション結果	65
3.4	実験結果	65
3.5	まとめ	67
第4章	極値近傍探索アルゴリズム	69
4.1	概要	69
4.2	極値近傍探索アルゴリズムの構築	69
4.2.1	近似曲線の導出	69
4.2.2	極値の判定	70
4.2.3	次世代個体	72
4.2.4	シンプレックス交叉	74
4.2.5	極値近傍探索アルゴリズムの流れ	76
4.3	シミュレーション結果	77
4.3.1	シミュレーション結果	77
4.3.2	評価値が同程度の場合	78
4.4	実験結果	84
4.5	まとめ	90
第5章	多極集中型最適化アルゴリズム	91
5.1	概要	91
5.2	ランダム生成による探索点の初期配置	91
5.3	多極集中型最適化アルゴリズムの設計	94
5.3.1	分散制御アルゴリズムの設計	94
5.3.2	集中アルゴリズムの設計	95
5.4	2変数における最適化アルゴリズムの収束性能の比較	98
5.4.1	初期探索点の比較	98
5.4.2	収束性能と探索点配置の比較	98
5.5	最適化結果	101
5.6	実験結果	104
5.7	まとめ	106
第6章	クラスター分析を応用した空間探索アルゴリズムの構築	107
6.1	概要	107
6.2	クラスター分析	107
6.3	空間探索アルゴリズムの設計	109
6.4	探索結果	109
6.5	実験結果	112
6.6	まとめ	114

第7章 結言と今後の展望と課題	115
7.1 結言	115
7.2 今後の課題と展望	116
学術論文と国際会議	117
謝辞	121
参考文献	123
付録A CFDシミュレータ	131
A.1 基礎式	131
A.2 VOF法	131
A.3 固液相流動	132
A.4 メッシュ生成法	132
付録B 溢流抑制制御の搬送軌道に対する速度カーブの導出過程	135
付録C CFDシミュレータを援用した自動注湯の最適傾動速度の導出	137
C.1 緒言	137
C.2 実験装置	138
C.2.1 自動注湯機の構成と注湯指令入力方法	138
C.2.2 注湯機の動作解析と傾動速度モデル	139
C.3 CFDシミュレータの流動評価	140
C.4 遺伝的アルゴリズムを用いた最適注湯入力の導出	143
C.4.1 液面レベル制御による製品欠陥低減	143
C.4.2 最適化問題の定式化	144
C.4.3 最適化計算結果と湯流れシミュレーション	145
C.5 製品部を考慮した解析結果の評価	146
C.6 最適速度入力を用いた注湯制御実験	146
C.7 結言	147
付録D 多極集中型最適化アルゴリズムの検証	151
D.1 初期探索点配置問題における遺伝的アルゴリズムを用いた時の収束性能	151
D.2 引力アルゴリズムを適用した場合の探索点の配置	155
付録E 5変数における多極集中型最適化アルゴリズムの性能調査	161
E.1 ダイカストの射出速度最適化における収束性能の調査	161
E.2 テスト関数を使った性能調査	163
E.2.1 Schwefel関数	163
E.2.2 2変数における最適化結果	164

E.2.3	5変数における最適化結果	167
E.3	まとめ	168
付録F	ダイカストプランジャの最適速度入力に関する追加実験	169
図目次		175
表目次		181

第1章 緒言

1.1 研究背景

これまで流体の圧力や流れなどの物理現象を解析する場合、その物理現象を単純化し、モデルを構築して解析を行うことで様々な制御を行ってきた^{[1]~[25]}。例えば角柱容器に対して液体のスロッシング現象のモデル化を行い、その固有振動数を知ることによってスロッシングを抑制した液体の搬送制御を可能にしたものもある^{[26]~[28]}。またモデル化することで、リアルタイムでの制御も可能となるため非常に有効な手段である。しかしながら実際のプラントに適用する場合、その容器や型の構造や形状は複雑なものが多く、さらには対象とする液体の物性値によっても挙動が変わるため、容易に単純化・モデル化できない場合が多い。そのため研究段階で終わることが多く、実用化し難いという課題があった。

一方で、近年、コンピュータの低価格化と高性能化を背景に、物体周りの流れや熱の流れを解析する技術として数値流体力学（Computational Fluid Dynamics：以下 CFD と表記）^{[29]~[31]}が使われるようになった。CFD とは Navier-Stokes 方程式と呼ばれる運動方程式やエネルギー保存式、質量保存式などを連動させて解く技術であり、Fig. 1.1 に示すように良好に液体の挙動をコンピュータ上で表現することができる。またこの CFD シミュレーションを使用することにより容易に、

- 流れの可視化
- 理想状態でのシミュレーション
- 実際に測定が困難な流体现象の測定
- シミュレーションの再現性
- 生産準備段階におけるコストを削減

をすることが可能となった。特に、形状や構造に左右されず解析を行うことができるため、多くの分野で欠陥の解析や設計を行う際の支援ツールとして利用されるようになった。

例えば鋳造では高温高圧の溶湯を取り扱っており、これまで可視化や計測をすることが困難であったため、その溶湯の注湯入力や速度設定を作業者の経験と勘によって行われていたものが、擬似的に流れを再現できる CFD シミュレータを利用したことで、それらの速度設定や入力に対する流体の挙動を知ることができ、そこから欠陥原因の解明やその欠陥の対策が容易になった。

このような CFD シミュレータを活用した研究が進むにつれ、単に欠陥の原因を解明するだけでなく、実験値とシミュレーションの流れの整合性を高めることや、その CFD シミュレータから得られた情報を基に、式を定義して評価方法を構築するなど、シミュ

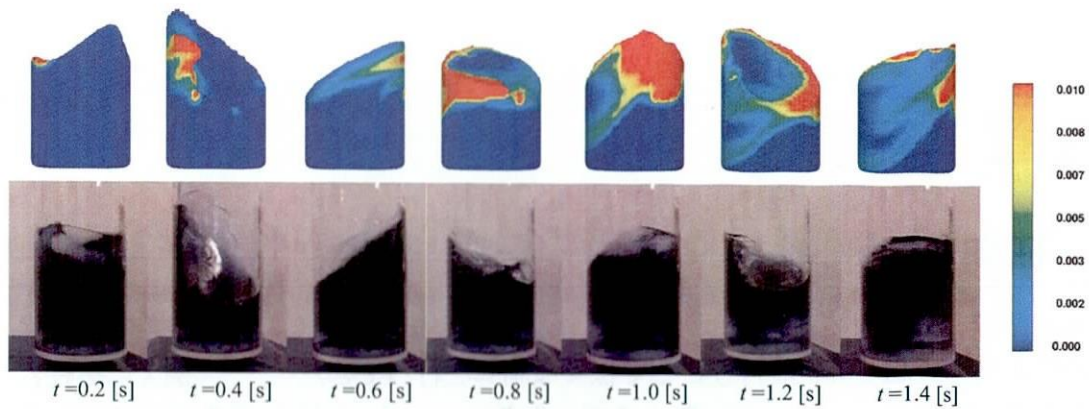


Fig. 1.1 Comparison of the fluid behavior between experimental result and simulation result

レーションを用いた評価・妥当性が向上した。さらに最近では、コンピュータの処理能力が向上したため、これまで解析時間に2ヶ月や3ヶ月といった長い解析期間が必要だったのに対して、1時間程度で解析を終えるようになるなど大幅に短縮されたことから、CFDシミュレータを設計の支援ツールとしてではなく、それを援用して最適化を行うといった試みがなされるようになった。

1.2 従来研究

1.2.1 CFD シミュレータを用いた研究

CFD シミュレータを用いた研究として、天野らは流体のスロッシングにおける研究を行った。2次元の境界要素法を用いたスロッシング解析手法と渦点法を組み合わせたシミュレーション手法を開発し、く形容器内で水平方向の流入噴流が誘起する、自由液面のスロッシング現象の解析をした。解析を行った結果、水位と流量の違いで振動を起こすパターンが大きく3つに区分されることがわかった^[32]。

田中らはテラー級数を有限項で打ち切ったときの打ち切り誤差を評価することによって、毎時刻、最適な時間増分を決めることができ、大振幅の液体運動に対して安定で精度の良い数値計算をすることができることから、この方法を拡張した、3次元軸対称貯槽内の大振幅スロッシングの解法を構築した。結果、微小振幅振動に対する理論解や実験値との比較を行い、実用上十分な精度を有することを確認した^[33]。

寺嶋らはCFD シミュレータを用いて振動抑制を伴った溶湯の搬送制御の最適化を行った。具体的には、1次モードの振動を解析するために、これまでバネマスダンパで表されていた液面振動をCFD シミュレータを用いて表現するとともに、白色のノイズの振動を与えたときの液面の高さを知ることによってその固有振動数の解析を行った。結果、理論値とほぼ同じ固有振動数を算出することができ、液体の励起を抑制した搬送入力求めた結果、液面を揺らすことなく搬送を行えていることが確認できた^[34]。

鋳造分野においては、鋳造品の品質と生産効率の向上を目的として鋳造CAEを活用した研究がなされた^{[35]~[44]}。ダイカストにおいては、鋳造CAEを使用することにより凝固や湯流れ、熱伝導などの複雑な解析が可能となることを利用し、恩田らはアルミニウム合金ダイカスト部品について、湯流れ模様やポロシティを意図的に発生させた素材を用いてX線CT調査によりポロシティの状態変化を解析し、鋳造CAEの妥当性を評価することに成功した^[45]。ここでダイカストとはFig. 1.2に示すように、プランジャで型に溶湯を押し込んで製品を生産する鋳造方法のことを指す。

坂村らはダイカストにおけるランナー形状やゲートの方案設計を行う上で、ランナーの扇型角度やランナーのランド角度、ランド長さおよびゲート角度等が湯流れ挙動にどのように影響するかを研究した^[46]。解析ソフトにはADSTEFANを用いて行った。ゲート幅0.04[m]の直線ランナーをベースとして、4種類のランナー形状、3種類のゲート幅、4種類の射出速度条件を用いて解析を行った。主立った結果として、扇形ランナーの場合射出速度を0.45~0.60[m/s]以下にしないとランナー部で空気巻き込みが発生することがわかった。

飯見らは鋳造CAEのダイカスト湯流れ解析結果を可視化検証実験を行うことにより、解析精度の向上を行った^[47]。実験装置として可視化できるよう高温高压に耐えるガラス窓を型背面に取り付けるとともに、ハイスピードカメラにて実際の溶湯挙動を確認した。その結果、実機の湯流れパターンをシミュレーションで正確に再現可能とした。

三中西らは射出スリーブ先端で跳ね返ってくる波の形状も含め、巻き込みの少ない

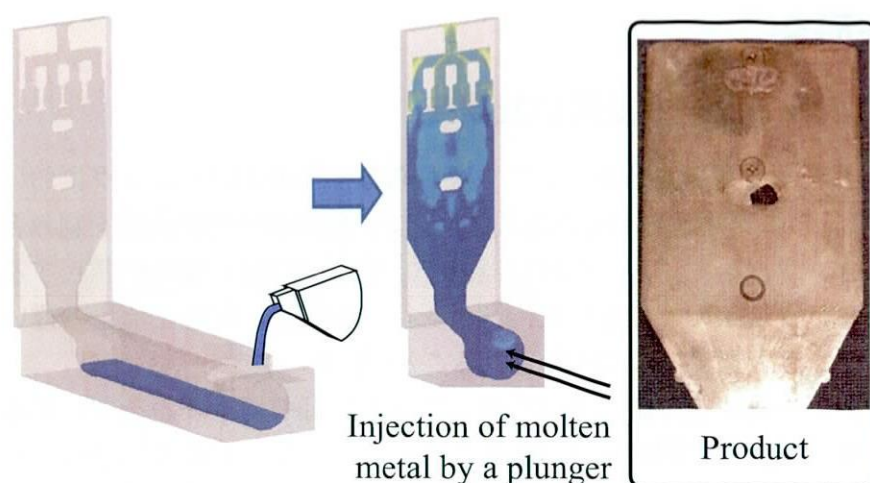


Fig. 1.2 A conception diagram of die casting and CFD simulation

条件や溶湯の先走りを検討することのできるシミュレータの開発を行った。結果、プランジャチップの移動に伴い仮想流体を均一に流入させることにより、簡易的に射出スリーブ内湯流れシミュレーションを可能にした^[48]。

加えて、重力鑄造においては、木間塚らは直交差分法により鑄型内背圧を考慮し、ガス欠陥を予測する手法を提案した^[49]。また、開発したコードをいくつかの湯流れ問題に適用した例を示し、X線湯流れ直接観察との比較および実製品への適用によって本解析手法の妥当性を検討した。主な結果として、検証実験を行った結果、充填パターンは良好に一致し、また圧力計測結果も一致したことからコードの妥当性を確認することができた。

1.2.2 最適化アルゴリズムの研究

一方で、最適化に関する研究としては、広大な解空間を効率よく探索する方法として、遺伝的アルゴリズムのような発見的手法であるメタヒューリスティックアルゴリズムがよく用いられる^{[50]~[54]}。特に最近では優良解を導出する局所探索法と多様性を維持し、高精度な解を導出するメタヒューリスティクス両者の利点を複合した手法が提案されている。

通常の遺伝的アルゴリズムに局所探索法を組み込んだハイブリッド型のアルゴリズムとして、遺伝的局所探索法 (Genetic Local Search : GLS) がある。この手法は、遺伝的アルゴリズムにより解析した集団の中から最良のものを選択し、その近傍で得られた個体を次世代集団の数個体と入れ替え解析をする。これは大域的探索と局所的探索を交互に繰り返す手法であり、その効率性や有効性も立証されている^{[55][56]}。

昆虫の行動を基に構築した最適化アルゴリズムとしてアントコロニー最適化アルゴリズム^[57]がある。このアルゴリズムは蟻の行軍の要因であるフェロモンの概念を取り

入れ、そのフェロモンバランスによって良好な経路を探索していくアルゴリズムである。また遺伝的アルゴリズムやアントコロニー最適化アルゴリズムなどの進化的アルゴリズムに局所探索法を組み込んだ複合的アルゴリズムは単純に「Memetic アルゴリズム」と括られ、今日の進化的アルゴリズムのほとんどがこの流れを組むものである^[58]。

近似曲線などの関数を用いて解空間を予測し、探索を行っているアルゴリズムとして SQP (Sequential Quadratic Programming)^[59] や EGO (Efficient Global Optimization)^[60] がある。具体的には統計学に基づき変数と評価の関係について調べ関数化すると共に、最小解があると予測される空間へと探索点を配置する手法である。

さらに最適化問題の一つに局所解と呼ばれるものがある。局所解とは凹凸のある解空間の中で、最大化問題の場合は凸状となっている部分を、最小化問題の場合は凹状となっている部分を局所解 (局所最適解) と呼び、その局所解の中でも、もっとも評価が良好なものを大域的な最適解と呼ぶ。一般的に最適化は勾配を調べ、その情報から良好な空間へと探索箇所を移動させていき最適解を見つける。しかしながら局所解をもつ解空間の場合、勾配法のための探索では局所解に陥った場合、探索点がそこから抜け出せ難くなり、結果、その局所解を大域的な最適解であると誤認してしまう。このように局所解の存在する空間では勾配法だけでは解くことが難しい。その一方で、メタヒューリスティックに基づいた探索方法では局所解への陥りを抑制するため、突然変異という概念を取り入れている。しかしながらその設定は難しく、突然変異率の確立を上げることは、局所解への陥りを抑制する一方で収束性能の低下を招く。そこで局所解をもつ解空間で、より効率良く探索を行うために一つ一つの有用な解を局所探索し、大域的な最適解の有無により探索空間を狭めていく手法や、大域的最適解近傍と推測される領域を連鎖的に探索する手法が提案されている^{[61]~[63]}。

1.2.3 CFD シミュレータを援用した最適化の研究

CFD シミュレータを援用して最適化を行った研究としては、Harvey らは牽引する家畜トレーラにおいて、牽引トレーラの屋根前方の形状は空気抵抗を減らすための重要な要素であり、また動物が入る内部空間の環境を改善するには、中と外との換気が非常に重要であることから、動物の居住空間の改善と牽引の抵抗を軽減させるために家畜トレーラの最適形状を算出した^[64]。結果、形状を最適化したことで牽引の抵抗が減ると共に、トレーラ内部の温度や湿度を削減することができた。

桜木らは遺伝的アルゴリズム^[66]を用いて湯境低減のためのダイカストにおける射出速度の最適化^[67]を行った。湯境とは、溶湯が型に流れ込む際の溶湯合流時に、溶湯先端が凝固していると融合することができず、結果、製品に境目ができたような状態になることを指している。これは鑄込み温度が低すぎるのも原因となるが、溶湯の充填時間が長いことが主な問題である。その一方で、不用意な高速射出の設定は空気の巻き込みが起り易くなり、製品内部に気泡が形成されてしまう。そこでシミュレータ

を用いて湯境の評価手法を構築し、湯境が起こらない程度でゆっくりと充填する射出速度の算出を行った。

齋藤らは凝固シミュレーションとタグチメソッドによる鑄造条件の最適化を行った^[68]。タグチメソッドとは工業分野における良品の生産確率を向上させるための考え方のことを指す。具体的には、良品条件を得られる解を単に求めるのではなく、品質のばらつきを少なくする条件の発見を第一に行い、そのばらつきの少なくする条件を基に、目標の良品条件を得られる解へと近づけていく手法である。結果、温度勾配が安定し引け巣の欠陥低減をシミュレーション上で確認することができた。

空気巻き込み低減可能なダイカストにおけるプランジャ速度最適制御に関する研究として、高橋らは最適化ソフトウェアである iSIGHT^[69]と CFD シミュレータを連携させることにより最適製造条件を自動的に導出した^[70]。最適化において遺伝的アルゴリズムを始め、近似手法や実験的手法などもソフトには用意されており、ユーザーがいずれかの最適化手法を決定して最適化を行うといったものである。

本研究室においては、バイオ分野で行われている攪拌操作の一つである揺動攪拌において、高速に、かつ空気巻き込みを軽減させる攪拌法の開発を行った^[65]。液面振動抑制制御として、ハイブリッド整形法^[28]を適用した。ヨウ素の脱色反応を利用した脱色法攪拌実験で検証を行ったところ、攪拌時間が無制御に比べ、液面の振動が抑制され、液面の崩れ及び空気巻き込み量が減少することを示し、制振制御の有効性を確認した。

また、ダイカストにおける品質向上を目的としたダイカストチップの形状最適化を行った^[71]。速度だけでなく、形状も空気巻き込みによる欠陥の抑制に効果があることが考えられてきたが、実際に最適化を行った結果はなく、そのため現在でもチップの先端形状は平型か、凸型の2種類しかなかった。そこで CFD シミュレータを援用した形状最適化手法を構築し、プランジャチップの最適化を行った。結果、最適化されたプランジャチップを用いることで、品質を向上させることができた。

1.3 CFDシミュレータを援用した最適化における問題点

CFD シミュレータを援用した最適化における問題は大きく分けて3つ存在する。その一つ目は、局所解が多数存在する空間 (多峰性解空間) の問題である。一般的な多峰性解空間の例として Schwefel 関数の解空間を Fig. 1.3 に示す。図から、多数の凹凸が見られ、多くの局所解が確認できる。一方で、CFD シミュレータから得られる評価が形成する解空間の例として、ダイカストにおけるプランジャ速度の各変数とその評価値が形成する解空間を Fig. 1.4 に示す。ここで Fig. 1.4 は評価値の低いものほど優良解であることを示している。Fig. 1.4 の解空間は実線で示した箇所のように、大きく二つの局所解が存在していることがわかる。しかし破線で示した箇所には多数の小さな凹凸を持つ解空間となっており、ここでも多峰性解空間を形成していることがわかる。これはシミュレーションの計算過程における、桁落ちや情報落ちなどの計算誤差が主

な原因と考えられる。このように CFD シミュレータを援用した最適化を行う場合、通常の最適化問題に加え、計算誤差などで形成された多峰性解空間の問題を解かなければならない。

二つ目は計算時間の問題である。従来の最適化手法では、一回の解析に 1[ms] 程度の非常に短時間で解ける問題を対象としていた。しかしながら、CFD シミュレータでは一回の解析に数分～数時間の計算時間が必要となる。そのため、従来の最適化手法で要した計算回数では有限時間内に解析を完了することができず、探索点や計算回数の削減をしなければならない。その一方で、前述の問題点である多峰性解空間の問題を解く場合、局所解に陥る確率が高くなり、大域的な最適解が見つかり難くなる。

三つ目は最適解が算出されても実験で良好な結果を得られない場合があることである。例えば、Fig. 1.3 における解空間では、各局所解のもっとも評価の良好な箇所は、勾配が 0 となる各局所解空間の中心点であり、同時に大域的に良好な評価を得ていることになる。しかしながら CFD シミュレータからの評価を用いた解空間では、Fig. 1.4 の破線に見られるように、多数の小さな局所解が存在するため、勾配が 0 となる箇所も多数存在することになる。そのため大域的に良好な評価を得ている解を見つけることが困難となり、結果、実験で良好な結果を得られない解を算出してしまう。

そのため CFD シミュレータを援用した最適化では、シミュレータの評価から実験対象に対して最適解を得るための最適化手法の構築も然ることながら、少ない探索点数・計算回数でも効率良く探索を行うとともに、実験でも良好な結果を得られるような解を算出する最適化アルゴリズムの構築をする必要がある。

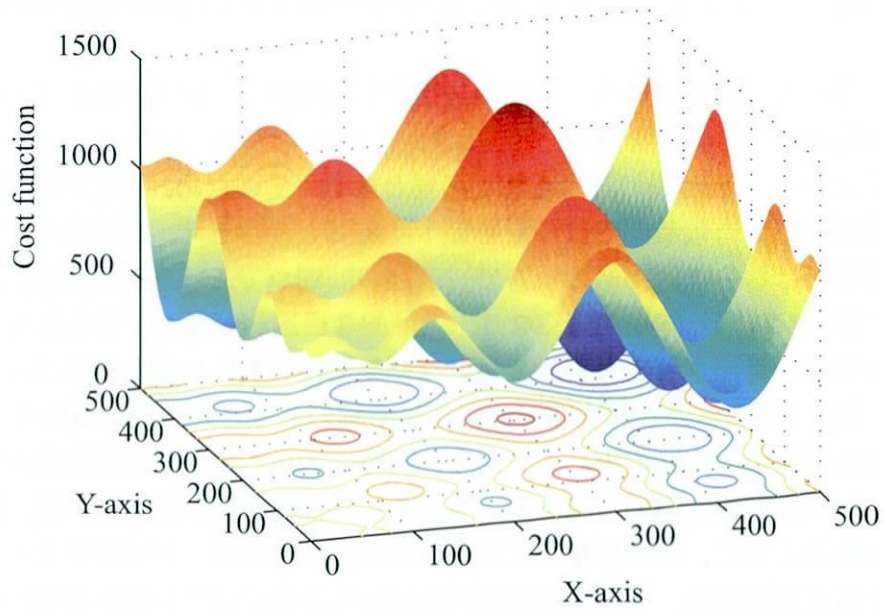


Fig. 1.3 Multimodal search space

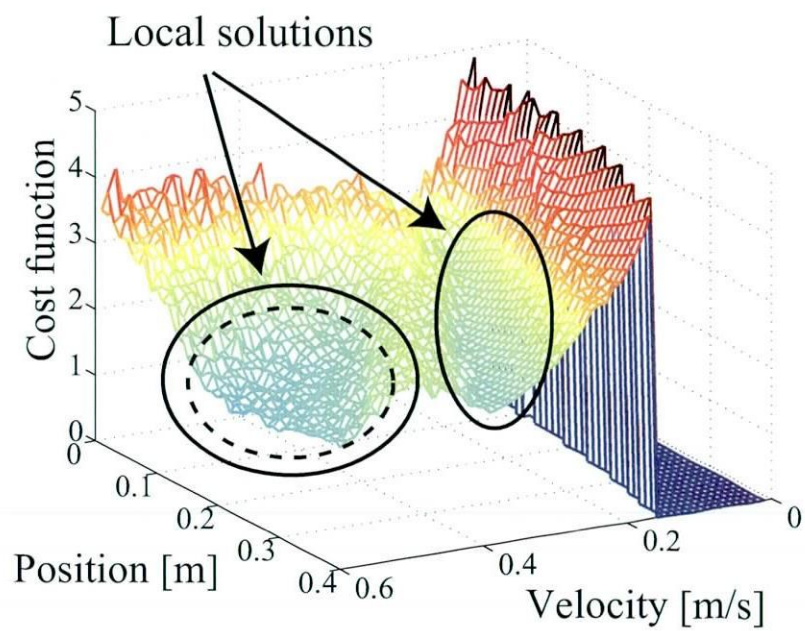


Fig. 1.4 Cost function of die casting process

1.4 研究目的と構成

本研究ではCFDシミュレータを援用した最適化手法の実用化を目的として、6自由度マニピュレータを用いた溢流抑制制御と、ダイカストにおける空気巻き込み低減を目指したプランジャ速度の最適化手法を構築するとともに、少ない計算回数かつ実験でも良好な結果を得られる解を算出することのできる最適化アルゴリズムの構築を行う。具体的には一般的に広く使われ、多様性に富んでいる遺伝的アルゴリズムをベースにして、近似曲線を援用することで極値を選定、大域的に良好な評価を得ている解を算出することのできる極値近傍探索アルゴリズム (Extremal Distribution Sorting Algorithm: 以下EDSA) を提案する。もう一方で、探索点の配置問題に着目し、分散制御則を利用して探索点の配置を行う多極集中型最適化アルゴリズム (Multi-subcenters Solution Search Algorithm) を提案するとともに、多極集中型最適化アルゴリズムに、大域的に良好な評価を得ている解の導出を可能にする、クラスター分析^{[72]~[74]}を応用した空間探索アルゴリズム (Space Analysis Algorithm) の構築を行う。最適化手法に関する有効性は実験を通して示し、最適化アルゴリズムに対しての有効性は比較対象として良く用いられる遺伝的アルゴリズムと比較することで示す。

本研究は、CFDシミュレータを援用した最適化理論の研究であり、以下のような学術的新規性、独自性がある。

1. 溢流抑制制御手法の提案
2. ダイカストにおけるプランジャの最適入力速度導出法の提案
3. 多峰性解空間に対する解の予測手法の提案
4. 実際のプラントにおいて良質な結果を得る解を算出できる最適化アルゴリズムの提案
5. 分散制御則を応用した探索点配置手法の提案
6. クラスター分析を応用した空間探索アルゴリズムの提案

本論文は以下のように構成する。

第1章では、本研究の背景と目的を述べる

第2章ではCFDシミュレータを援用した最適化である、6自由度マニピュレータを用いた溢流抑制制御について述べる。スプーンに入った水を溢さずに搬送するため、シミュレーションにおける流体の挙動を実験結果と一致させた上で、溢流を抑制する搬送軌道を算出する。本手法の有効性を示すための比較対象として、同じ搬送時間となるよう設定した比例制御による搬送、そして振動抑制を伴った搬送方法の一つであるハイブリッド整形法を用いたときの搬送を行う。

第3章では、CFDシミュレータを援用した最適化の別例として、ダイカストにおけるプランジャ速度の最適化について述べる。この章では、CFDシミュレータを用いてダイカストの欠陥原因を解析すると共に、ダイカストプランジャの射出速度最適化について述べる。具体的に、実験とシミュレーションを比較検証し、欠陥要因を調べ、欠陥評価を含んだ最適化問題に定式化し、遺伝的アルゴリズムを用いて、そのプランジャ

速度の最適入力値を導出する。本手法の有効性は実験を通して示す。また、第2章を通じてCFDシミュレータを援用した最適化に求められる最適化アルゴリズムの性能について述べる。

第4章では、実験でも良好な結果を得られるような最適化アルゴリズムである、極値近傍探索アルゴリズムについて述べる。CFDを援用した最適化において、最適化を行っても実験では良好とならない場合ある。これは解析時間の問題により、探索点や繰り返し計算回数を減らしたことで局所解に陥り易くなっていることが原因であると考えた。そこで近似曲線を用いて、得られた評価値から近似曲線を生成し、その近似曲線が予測する良好な解空間と、その極値を判別することで、大域的に評価が安定となっている解空間を集中的に探索、最適解を導出する最適化アルゴリズムを提案する。有効性はダイカストにおけるプランジャ速度の最適化を行い、実験により示す。また最適化アルゴリズムの性能検証としてSchwefel関数を用い、本手法の有効性を示す。

第5章では、収束性能の向上と繰り返し計算回数の削減を目的とした分散制御則を応用した最適化アルゴリズムである多極集中型最適化アルゴリズムについて述べる。CFD最適化問題に対して収束性能に大きな差を生む要因として探索点の配置があげられる。これまで使われてきたランダム生成による探索点の配置では、探索点数が少ないために配置に偏りが生じ、未解析領域が発生し易くなる。そこで多極へと探索点を分散・集中させて探索することのできる最適化アルゴリズムを提案した。探索点を偏り無く配置させるための探索点分散法として、被覆制御の考え方を応用し、探索点自身が移動して他の個体との間隔の調整を行うようにする。空間全体への探索点の分散後、探索点周辺に既に評価されている箇所がある場合、その評価を参照して、探索点同士の間隔を保ちつつ移動する法則を探索点に与える。これにより解の向上が見込まれる空間へと探索点を集中させ、探索点の配置が多極化し、収束性能の向上や局所解への陥りを抑制する。有効性はダイカストにおけるプランジャ速度の最適化を行い、実験により示す。

第6章では、近似曲線を用いず大域的に良好な評価を得ている解を見つけることのできる空間探索アルゴリズムを提案する。実験でも良好な結果を得るためにはシミュレータの評価変動の影響を考慮しなければならない。そこで評価変動の影響を考慮した評価方法を構築することを目的とし、クラスター分析を応用して探索点の密度で良好な空間を判断することのできるアルゴリズムを提案する。有効性はダイカストにおけるプランジャ速度の最適化を行い、実験により示す。

第7章で結言と今後の展望を述べる。

第2章 6自由度マニピュレータを用いた 溢流抑制制御

2.1 概要

多くの国で少子高齢化社会を迎えつつあり、近い将来、サービス業や医療・福祉の分野において、労働力不足が非常に大きな問題となることが予想される。これらの問題を解決するために、ロボットによって労働力不足を補おうとする試みが多くなされている^{[75][76]}。とりわけ福祉分野において、介護者の負担を軽減させるためのアシスト装置や被介護者の自立を支援するための自立支援ロボットの研究が活発に行われており、特に被介護者の自立を支援するための自立支援ロボットは日常生活への復帰が可能となるだけでなく、福祉分野における労働力不足も解消する手立てとして大変期待されている。

その中でも食事は、毎日欠かすことのできない作業であるうえ、介護者、被介護者ともに非常に負担のかかる作業であることから、これまで Handy-1^[77]や My Spoon^[78], Winsford Feeder^[79], Neater Eater^[80], Bestic^[81]など様々な食事支援ロボットが研究・開発されている。しかしながら、それぞれのロボットの構造的や機能的問題から、食事できる品目が制限され、また食事完了までに必要以上の時間を要し、さらには食事中に食べ物をたくさん溢してしまうなど多くの問題をかかえている。特に、これまでのロボットでは、スープや味噌汁、おかゆなどの液状の食物を溢さずに迅速に運び、食事することは不可能であった。しかし利用者の中には、嚥下が困難である方や吸引力が弱い方も多く、スプーンで液状の食物の食事支援を可能とする必要がある。

一方で、これまでに液体の搬送制御として、液体の振動を抑制しながら、高速に搬送することを可能とする研究が多く行われている^{[82] ~ [85]}。しかしながら容器一杯に液体が満たされている場合には、ハイブリッド整形法^[85]などの制振制御を適用しても、加減速時の液面の傾動により、液体が溢れる可能性がある。さらにスプーンなどの小型かつ複雑な形状の容器を用いた場合、液面の振動現象は従来までに使用されていた振り子モデルなどでは厳密に表現することは困難である。

そこで本研究では、制御系設計に流体解析シミュレータを援用することにより、小型容器を用いた場合の搬送時の溢流を解析し、掬った液状食物を溢さずに搬送することを可能とする液体ハンドリング制御システムの開発を目的とする。本研究が実現すれば、食べ物の種類に制限なく、溢さず迅速に食事を支援することを可能とし、より実用的な食事支援ロボットを実現することが可能となる。さらに嚥下が困難である方や吸引力が弱い方も利用可能となり、結果として、食事支援ロボットの利用者層を広げ、少子高齢化の労働力不足問題に対しても貢献できる。

2.2 実験装置

本研究では, 6自由度マニピュレータを用いてスプーン上の液体の挙動解析と搬送実験を行う. 6自由度マニピュレータは Fig. 2.1 に示す安川電機製マニピュレータを使用した. 本実験装置の仕様を Table 2.1 に示す. 搬送容器には日常生活で一般的に使用されている形状のスプーンを用いる. 今回用いるスプーンの形状と寸法を Fig. 2.2 に示す.

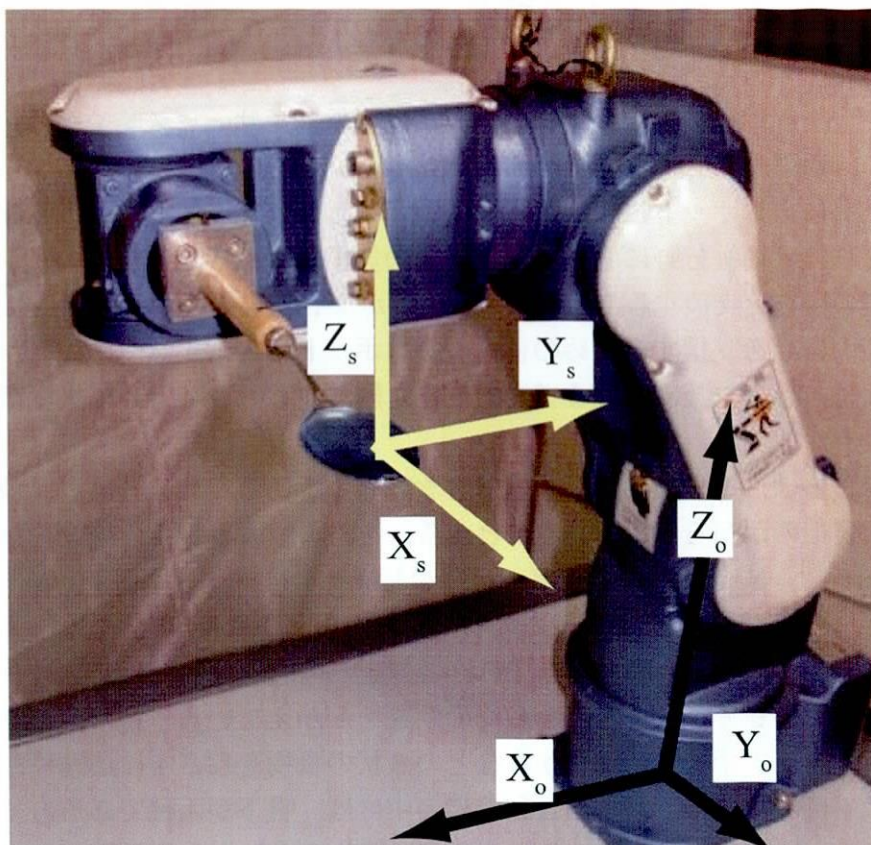


Fig. 2.1 Picture of 6DOF manipulator

Table 2.1 Maximum angular velocity of the motors

	Maximum angular velocity
1st motor	± 200 [deg/s]
2nd motor	± 128 [deg/s]
3rd motor	± 190 [deg/s]
4th motor	± 210 [deg/s]
5th motor	± 210 [deg/s]
6th motor	± 340 [deg/s]

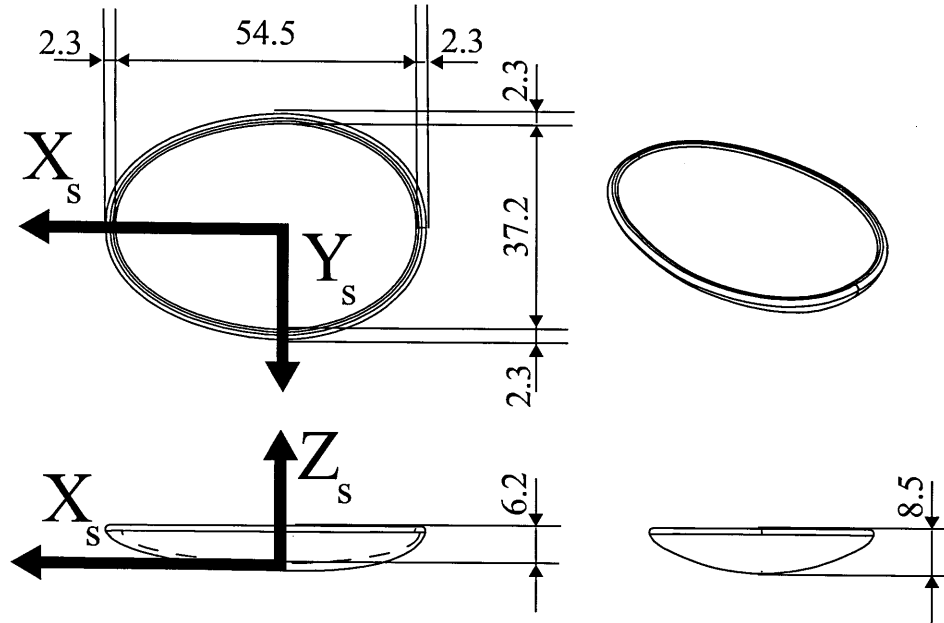


Fig. 2.2 Details of the spoon shape

2.3 CFD シミュレータの設定

本研究では，スプーン上の液体の挙動解析に流体解析ソフト *FLOW-3D*^[93]を用いる．

まず始めに，シミュレータのメッシュ幅の設定を行う．シミュレータのメッシュ幅は細かいほど計算の精度が上がり，実際の現象に近い挙動を得ることができる．しかしながら，細かく設定しすぎれば解析に要する時間が莫大となってしまう．特に流体解析シミュレータを援用して最適化問題を解く場合には非常に多くの計算を実行しなければならない．本研究では解析時間を可能な限り短く，かつ液体の挙動が十分に表現できるようにシミュレーションの解析領域を Fig. 2.3 のように設定した．そのときのシミュレーションの解析領域とメッシュ幅のパラメータを Table 2.2 に示す．このときの総メッシュ数は 175000 であり，解析時間は Pentium4 3.0GHz 搭載のパソコンで約 15 分程度であった．

Table 2.2 Setting mesh parameters of the spoon

	X direction	Y direction	Z direction
Minimum width	-0.03[m]	-0.021[m]	0[m]
Maximum width	0.03[m]	0.021[m]	0.015[m]
Total cells	100	70	25
Cell width	0.006[m]	0.006[m]	0.006[m]

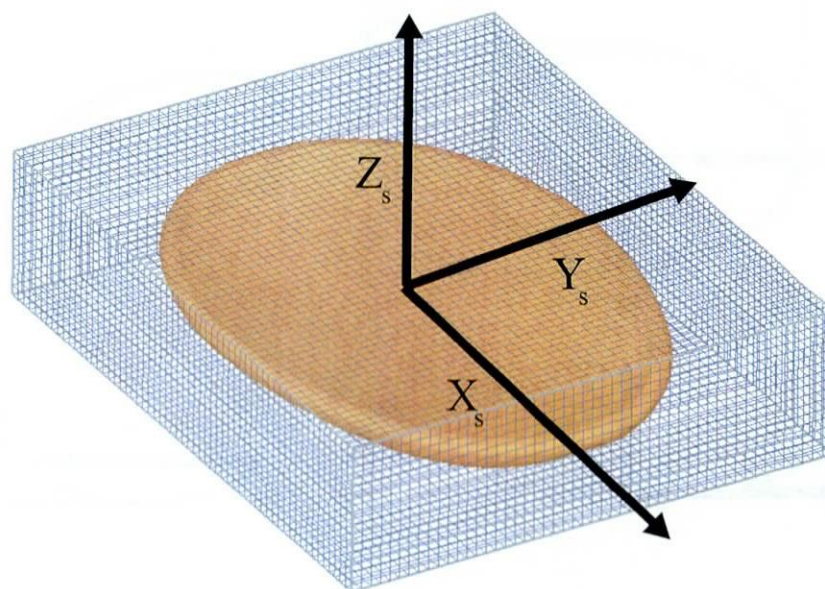


Fig. 2.3 Mesh setting of the spoon

対象液体は通常の水道水とし、液量はスプーンに擦り切れ一杯入る程度である 4.7×10^{-3} [ℓ] とした。解析時のシミュレーションパラメータを Table 2.3 に示す。ここで接触角とは、物質と液体の親水度を示し、接触角が大きいほど疎水性となり液体をよく弾く。また接触角が小さいほど親水性を示し、液体は物質に付着しやすい。つまり接触角が小さいほど溢れにくくなり、接触角が大きいほど溢れ易くなる。容器形状が複雑な場合、理論的に接触角を算出するのは困難となる。本研究では、シミュレーション解析を行なうことにより、接触角を決定した。

構築した液体解析シミュレータの有効性を検証するため、Fig. 2.4 の実線に示すような入力を用いて、 Y_s 軸方向への搬送実験および搬送シミュレーションを行った。こ

Table 2.3 Fluid parameters of water

Fluid parameters	Water
Density	1000 [kg/m ³]
Viscosity	100×10^{-5} [Pa·s]
Temperature of the Fluid	293.2[K]
Surface Tension	0.073 [N/m]
Surface Roughness	0.0025 [m]
Gravity	9.81 [m/s ²]
Contact Angle	75 [deg]

で Fig. 2.4 は、破線が参照速度カーブ、実線が実験における速度カーブを示している。その時の実験結果を Fig. 2.5 に、解析結果を Fig. 2.6 に示す。

結果として、液体の自由表面の挙動や水が溢れるタイミングを良好に表現しており、本シミュレータの有効性が伺える。また液体が溢れるという現象は、スプーンの加減速時に発生する液体の動きが表面張力などによる内向きの力を超えた場合に、起こっていることも確認できる。さらにロボットの最大加速度制約を満たした上で溢流を防止する搬送(加速)を行った場合、残留振動をほぼ起こさず、さらにスプーンは Y_s 軸に対しては対象であることから、溢さない加速度カーブが求まれば、減速時も同様の速度カーブを適用することで、溢さずに搬送を行うことができる。次節以降では、本シミュレータを援用し、液体ハンドリング制御システムを構築する。

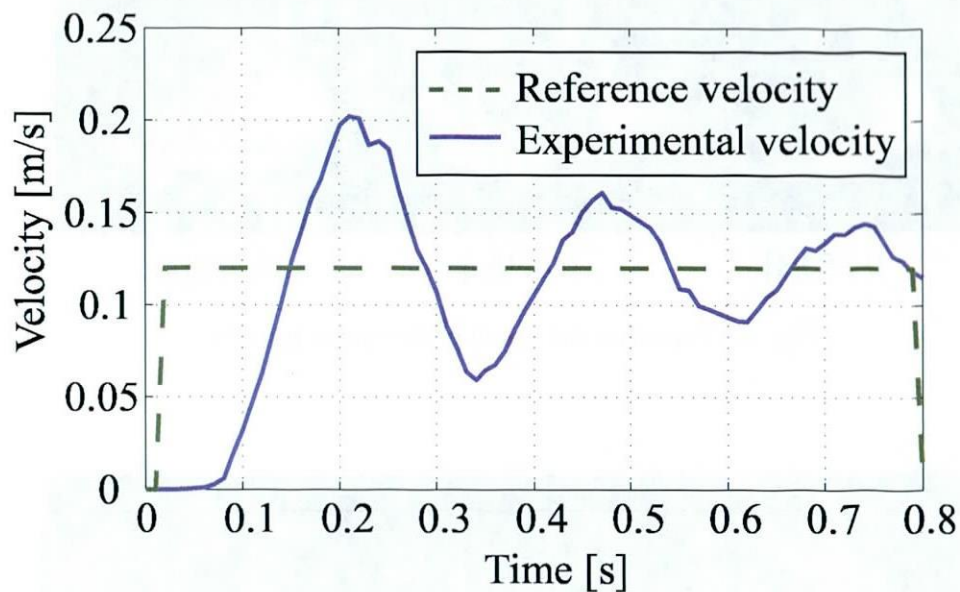


Fig. 2.4 Reference and experimental velocity of the spoon

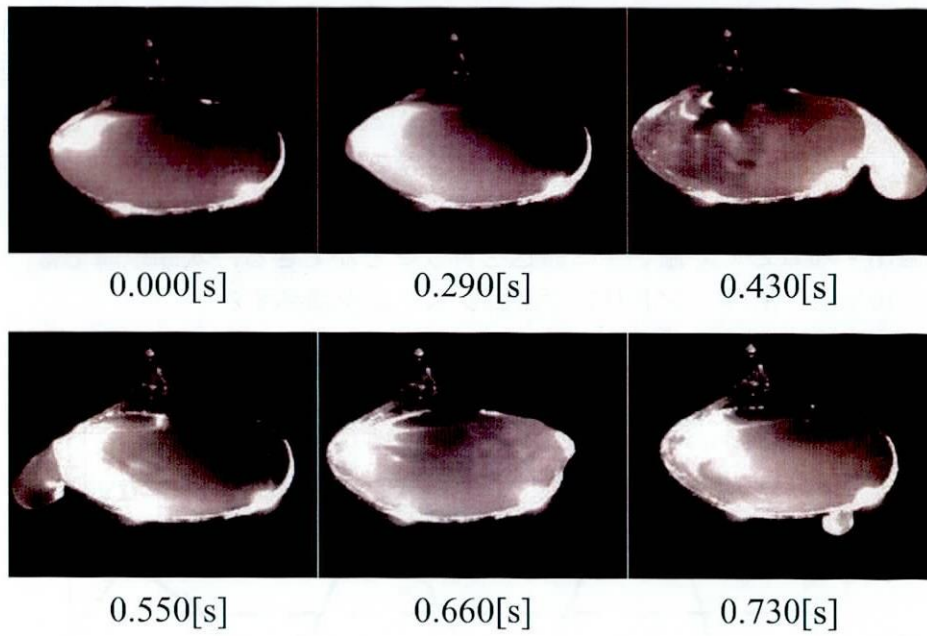


Fig. 2.5 Experimental result of the spoon transfer

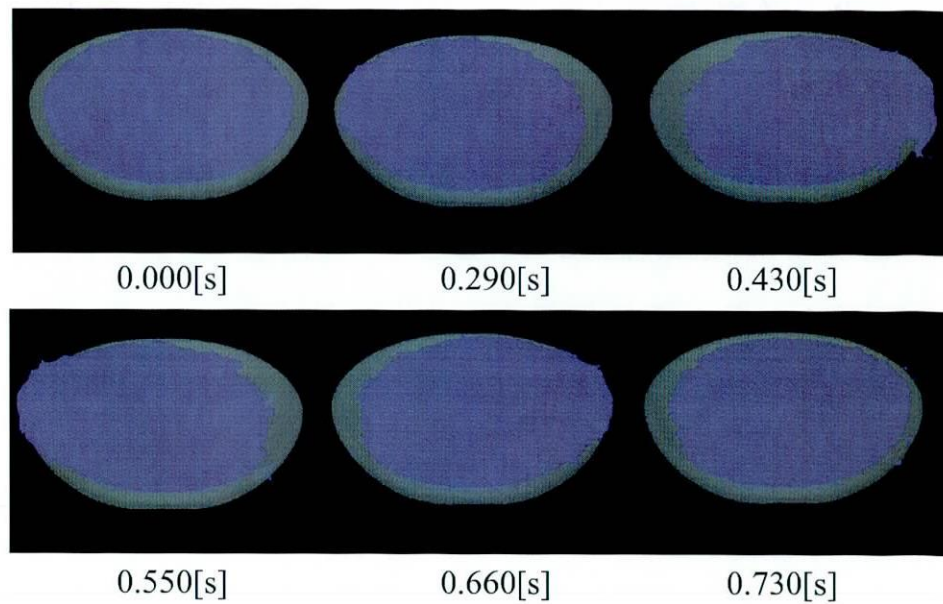


Fig. 2.6 Simulation result of the spoon transfer

2.4 モデリング

2.4.1 運動学

座標系の設定は様々な方法があるが, 本マニピュレータではD-H法をベースとして座標系の設定を行うとする. Fig. 2.7 にはその設定図を示し, この姿勢をマニピュレータの初期姿勢とする.

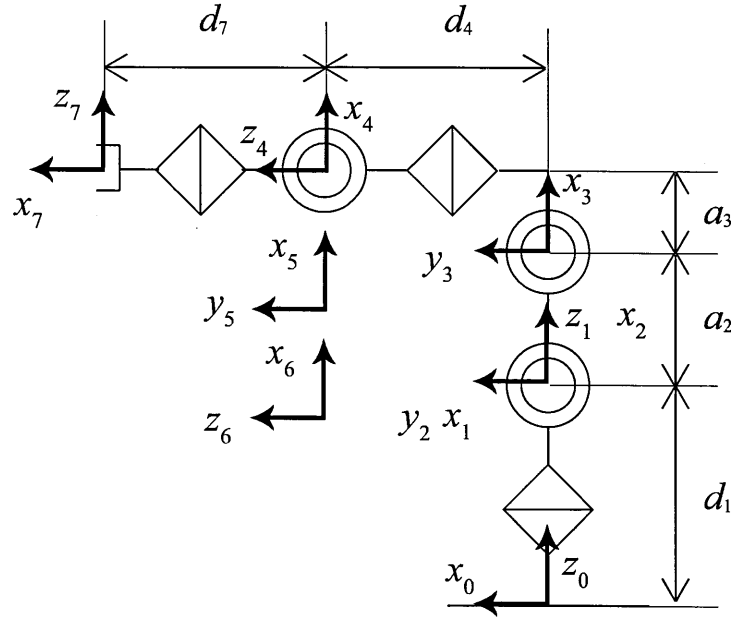


Fig. 2.7 Setup coordinate

各軸の座標系設定を以下に行う. 原点 Σ_0 はマニピレータの根元と台座の接地点中央に設定する. x_0 はマニピレータ横 (左) 方向, y_0 は手前方向, z_0 は垂直方向とする. Σ_1 を Σ_0 から d_1 だけ離れた第1, 第2関節の回転軸の交点にとり, z_1 を第1関節上に設定する. x_1 は第1, 第2関節の回転軸に直交するように設定する. Σ_2 の原点は Σ_1 と同じ原点に位置する. z_2 は第2関節回転軸と同じ向きにとり, x_2 は第2リンクと同じ向きである. Σ_1 と Σ_2 が同じ原点なので, $a_1 = d_2 = 0$ となる. リンクねじれ角は, z_2 が Σ_3 の原点は第3, 第4関節の回転軸の交点にとり. z_3 は第3関節回転軸と同じで, x_3 は第3, 第4関節軸回転軸に直行するようにする. 続いて $\Sigma_4, \Sigma_5, \Sigma_6$ の原点の設定をする. それぞれ, 第5関節回転軸の交点で交わったところに原点をとる. z_4 は第4関節と同じ向き, x_4 は第4, 5関節回転軸に直交するようにとる. z_5 は第5関節と同じ向き, x_5 は第5, 6関節回転軸に直交するようにとる. Σ_4, Σ_5 は原点が同じなので, $a_4 = d_5 = 0$ となる. z_6 も同様に第6関節回転軸と同じ向き, x_6 も x_5 と同様の向きとおく. ここまで設定したリンクパラメータをまとめて Table 2.4 に示す. Σ_7 の座標系はマニピレータの手先位置からハンド部の長さとして将来的に持たせるであろうスプーンの長さを伸ばした先とする.

Table 2.4 Link parameter of 6DOF manipulator

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	d_1	θ_1
2	-90°	0	0	θ_2
3	0	a_2	0	θ_3
4	-90°	a_3	d_4	θ_4
5	-90°	0	0	θ_5
6	-90°	0	0	θ_6

運動学とは現在の各関節角度から位置・姿勢を得るものである。安川ロボット MOTOMAN-UPJ の運動学をに解く。先ほど得たリンクパラメータを使用し, Eq.(2.1) の座標変換行列 ${}^{i-1}T_i$ をかけると Σ_{i-1} から Σ_i が得られる。

$${}^{i-1}T_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

ベース座標系から, 手先の座標を得るには Eq.(2.2) で得ることができる。 0T_7 の 1 行 1 列から 3 行 3 列までは姿勢を表し, 4 行 1 列から 4 行 3 列までは位置を表す。

$$\begin{aligned} {}^0T_6 &= {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 \\ &= \begin{bmatrix} r11 & r12 & r13 & x \\ r21 & r22 & r23 & y \\ r31 & r32 & r33 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2.2)$$

$$\begin{aligned} r11 &= c1(c23(c4c5c6 - s4s6) - s23s5c6) + s1(s4c5c6 + c4s6) \\ r21 &= s1(c23(c4c5c6 - s4s6) - s23s5c6) - c1(s4c5c6 + c4s6) \\ r31 &= -s23(c4c5c6 - s4s6) - c23s5c6 \\ r12 &= -c1(c23(c4c5s6 - s4s6) - s23s5s6) - s1(s4c5c6 - c4s6) \\ r22 &= s1(c23(c4c5s6 - s4s6) - s23s5s6) + c1(s4c5c6 - c4s6) \\ r32 &= s23(c4c5s6 + s4s6) - c23s5s6 \\ r13 &= -c1(c23(c4s5 + s23c5) - s1s4s5) \\ r23 &= -s1(c23(c4s5 + s23c5) + c1s4s5) \\ r33 &= s23c4s5 - c23c5 \end{aligned} \quad (2.3)$$

$$\begin{aligned} x &= c1(a_2c2 + a_3c23 - d_4s23) \\ y &= s1(a_2c2 + a_3c23 - d_4s23) \\ z &= d_1 + (a_2s2 + a_3s23 + d_4c23) \end{aligned} \quad (2.4)$$

ここで, $c1$ は $\cos(\theta_1)$, $c23$ は $\cos(\theta_2 + \theta_3)$ と略記している. 基本姿勢で手先の姿勢をベース座標系と一致するようにする. Z-Y-Xオイラー角で表現すると, $\alpha = 180^\circ$, $\beta = 90^\circ$, $\gamma = 0^\circ$ である.

$$\begin{aligned}
 {}^0R_7 &= {}^0R_6 {}^6R_7 \\
 &= {}^0R_6 \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \\
 &= \begin{bmatrix} r13 & -r12 & r11 \\ r23 & -r22 & r21 \\ r33 & -r32 & r31 \end{bmatrix}
 \end{aligned} \tag{2.5}$$

Eq.(2.5) を計算することで, 初期姿勢をベース座標系に一致させることができる.

2.4.2 逆運動学

逆運動学はロボットを任意の姿勢や位置にするための各関節の移動量を知ることができる. 本研究で使用する安川ロボットのような6自由度マニピュレータの逆運動学は解析に解くことができる. 手首の位置を第5リンクの回転軸にとる. そうすることで, 姿勢決定に手先3自由度, 位置決定に根元3自由度と分割が可能となる. まず, 手首の位置 (x_r, y_r, z_r) がどこになるのかを求める. これは根元の3関節 $\theta_1, \theta_2, \theta_3$ より求めることができる. マニピュレータを真上から見下ろすと, 第3, 第4, 第5リンクは, X-Y 平面の原点を通り (x_r, y_r) に向かう線分として見えるので, この線分とX軸とのなす角が θ_1 となる. よって Eq.(2.6) のようになる.

$$\theta_1 = \text{atan2}(y_r, x_r) \tag{2.6}$$

次に θ_3 を求める. θ_3 は次式により与えられる.

$$\theta_3 = \text{atan2}(a_3, d_4) - \text{atan2}(\kappa, \sqrt{a_3^2 + d_4^2 - \kappa^2}) \tag{2.7}$$

ただし, κ は

$$\kappa = \frac{x_r^2 + y_r^2 + x_r^2 - a_2^2 - a_3^2 - d_4^2}{2a_2} \tag{2.8}$$

とする. θ_2 も

$$\theta_2 = \text{atan2}(c23, -s23) - \theta_3 \tag{2.9}$$

で表すことができる. $s23$ と $c23$ は次の式で表すことが可能なので, θ_2 が求まることになる.

$$\begin{aligned}
 L &= z_r^2 + (c1x_r + s1y_r)^2 \\
 s23 &= ((-a_3 - a_2c3)z_r - (c1x_r + s1y_r)(d_4 - a_2s3))/L \\
 c23 &= ((a_2s3 - d_4)z_r - (a_3 + a_2c3)(c1x_r + s1y_r))/L
 \end{aligned} \tag{2.10}$$

ここからは、姿勢による手先3自由度の解を求める。それぞれ、

$$\theta_4 = \text{atan2}(r_{11}s_1 + r_{21}c_1, -(r_{11}c_1 + r_{21}s_1)c_{23} + r_{31}s_{23}) \quad (2.11)$$

$$\begin{aligned} \theta_5 &= \text{atan2}(s_5, c_5) \\ s_5 &= -(r_{11}(c_1c_{23}c_4 + s_1s_4) + r_{21}(s_1c_{23}c_4 - c_1s_4) - r_{31}s_{23}c_4) \\ c_5 &= r_{11}(-c_1s_{23}) + r_{21}(-s_1s_{23} - c_{23}r_{31}) \end{aligned} \quad (2.12)$$

$$\begin{aligned} \theta_6 &= \text{atan2}(s_6, c_6) \\ s_6 &= -r_{13}(c_1c_{23}c_4 - s_1s_4) - r_{23}(s_1c_{23}c_4 + c_1s_4) + r_{33}s_{23}s_4 \\ c_6 &= r_{11}((c_1s_{23}c_4 + s_1s_4)c_5 - c_1s_{23}s_5) \\ &\quad + r_{21}((s_1c_{23}c_4 - c_1s_4)c_5 - s_1s_{23}s_5) - r_{33}(s_{23}c_4c_5 + c_{23}s_5) \end{aligned} \quad (2.13)$$

によって、目標姿勢位置に対して、各関節変数 θ_i が得ることができる。

逆運動学が正しいかを確認するため、初期姿勢 $(x,y,z,\text{roll},\text{yaw},\text{pitch})=(0.36, 0, 0.56, 0, 0, 0)$ から、目標姿勢を $(x,y,z,\text{roll},\text{yaw},\text{pitch})=(0.45, 0.20, 0.30, 0, 0, 0)$ として逆運動学を通して実際にマニピュレータで検証してみる。

Fig. 2.8, Fig. 2.9 は破線が逆運動学に通した場合のリファレンスの角度とそれをマニピュレータに通した時の角度応答である。第4軸と第6軸がリファレンスと現在角度が一致していないが、第4軸と第6軸はともに対象なりファレンスを与えており、実際の姿勢は水平を保ったままであることがわかる。その他の軸はリファレンスに対して良好に追従していることがわかる。Fig. 2.10, Fig. 2.11 は実験より得られた角度を順運動学に通したときに得られる位置 x,y,z と姿勢 $\text{roll},\text{yaw},\text{pitch}$ である。目標角度に向かって追従はしているが、途中で姿勢が崩れているところがあるが、これは先ほどの第4軸と第6軸の関係上、2軸が同期して駆動していない結果誤差が $\pm 5[\text{deg}]$ 程度発生してしまったと考えられる。しかし、位置に関してはリファレンスに対して追従しており、良好な結果が得られた。Fig. 2.12 は3次元 x,y,z の位置をプロットしたものである。初めのズレに関しては、サーボオンをした後にブレーキを開放するのだが、その時自重により位置ズレを起こしてしまう。この事を除けば、これらから逆運動学が正しく求まったことがわかる。

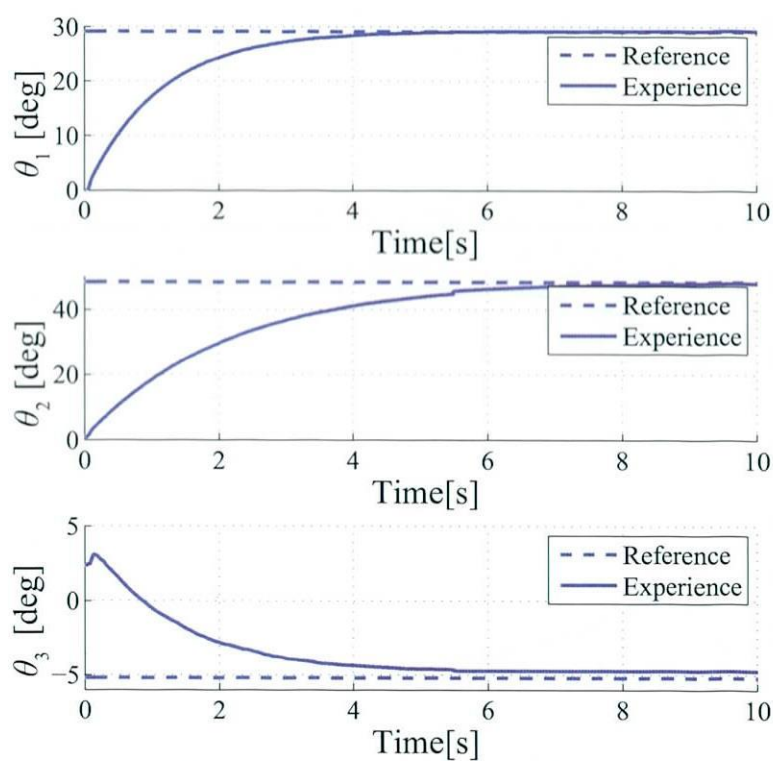


Fig. 2.8 Experimental result of angle

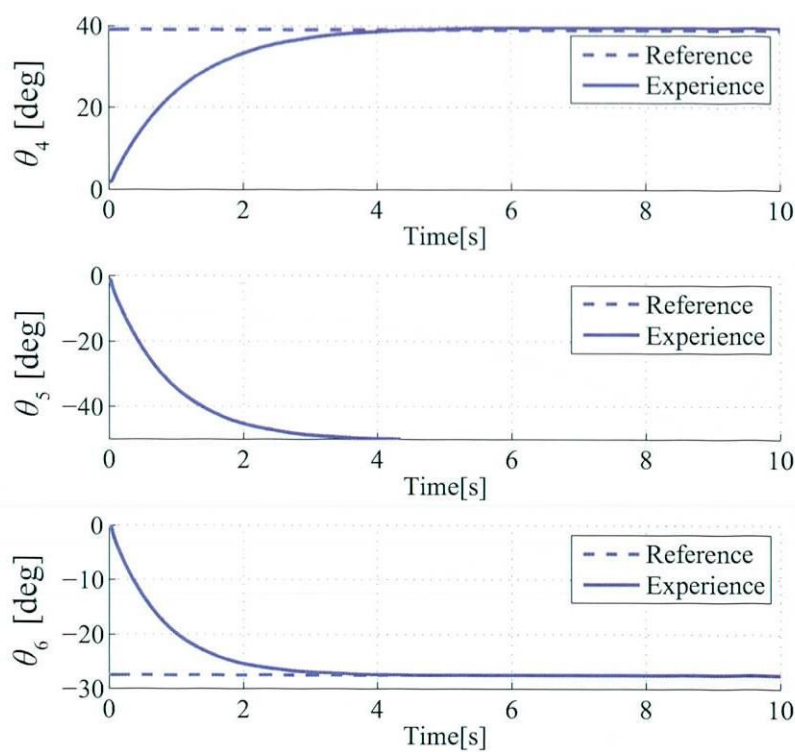


Fig. 2.9 Experimental result of angle

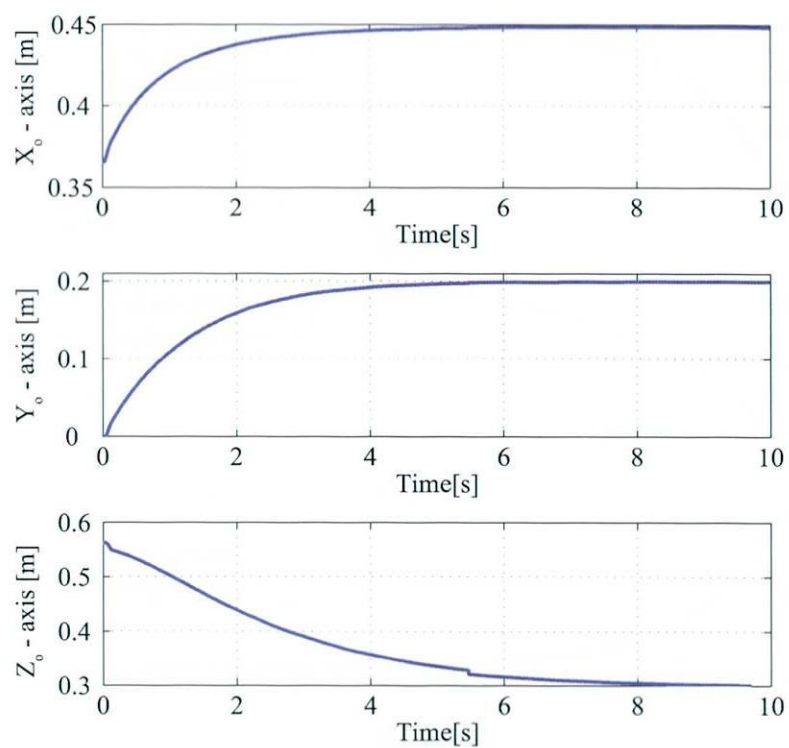


Fig. 2.10 Experimental result of position

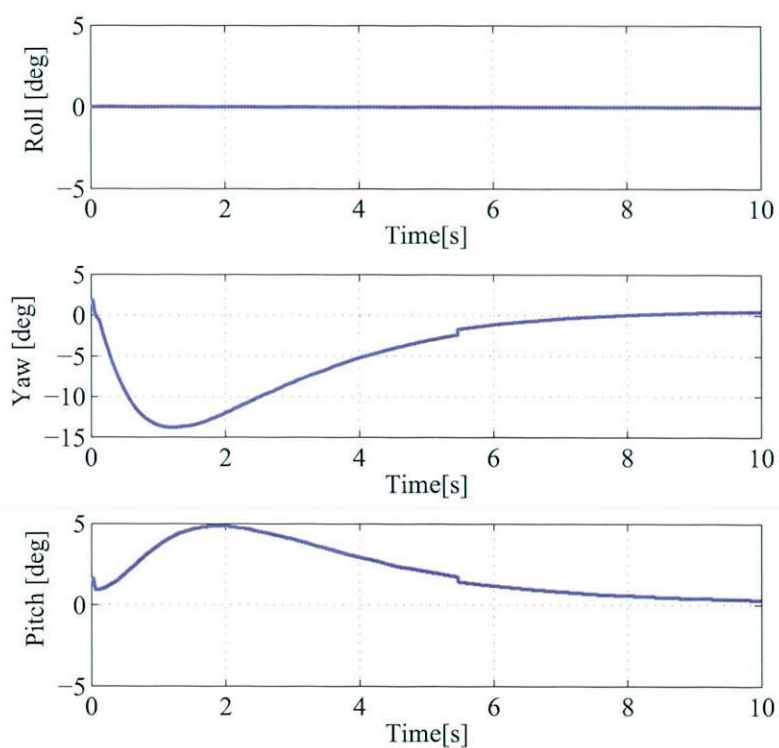


Fig. 2.11 Experimental result of orientation

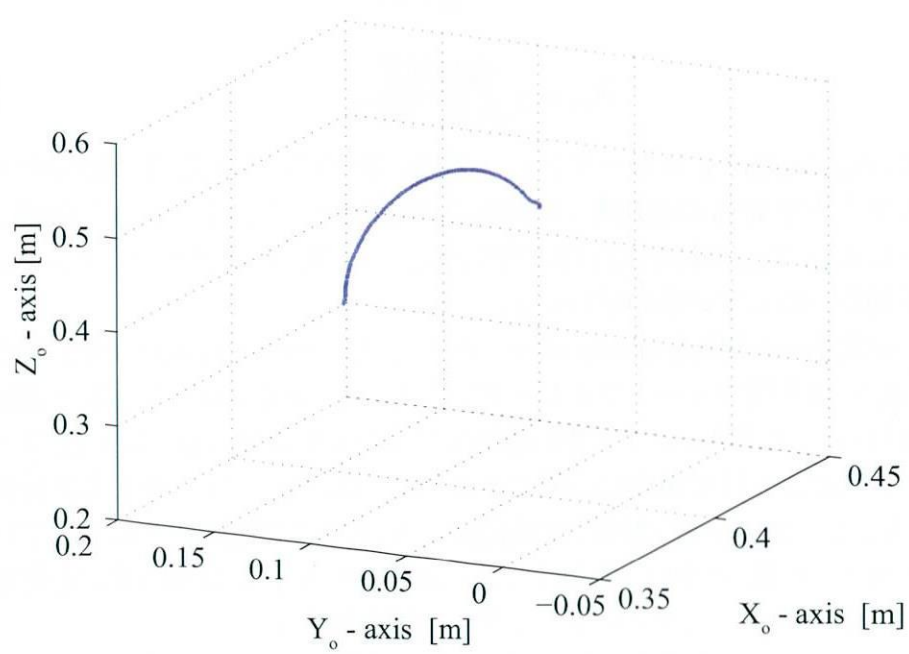


Fig. 2.12 Experimental result of position

2.4.3 モータモデルの同定

- 第1軸のモータ同定

マニピュレータのサーボアンプは *SGDS - A5F01A* と *SGDS - 0F01A* を使用し、モータは *SGMAH - 01A0A21* と *SGMAH - A3A1A21* を使用している。まず、同定する前にオフセットを取り除く必要がある。オフセットは 0[V] 出力したときに、サーボアンプ側でオフセットの調節を行う。サーボアンプの制御方法は速度制御で設定をした。速度制御とはある指令電圧 [V] に対して、ある指令速度 [min^{-1}] を出力する。ゲイン設定では 6[V] 入力で定格回転速度となる。DC サーボモータは Eq.(2.14) で表されるように、サーボモータへの入力電圧 $u(t)$ から回転速度 $\dot{\theta}(t)$ の伝達関数を 1 次遅れ系として近似され、さらに積分することで回転角度 $\theta(t)$ が得られる。

$$G(s) = \frac{K_m}{s(T_ms + 1)} \quad (2.14)$$

$$K_m = \frac{\text{搬送距離}}{\text{電圧値の積分}} \quad (2.15)$$

ここで、 $K_m[\text{deg}/(\text{s} \cdot \text{V})]$ はモータゲイン、 $T_m[\text{s}]$ は時定数である。第1軸のモータにステップ入力で 1[V] の電圧を 1 秒間与え、実験値とシミュレーション値が一致するように K_{m1} , T_{m1} の値を試行錯誤的に決定した。Fig. 2.13 よりシミュレーションと実験値が一致しているのがわかる。

次に同定結果の検証を行った。求まった K_{m1} , T_{m1} を用いて、電圧 1[V], 周期 $2\pi/3$ の \sin 波を 10 秒間フィードフォワードで入力して、シミュレーションと実験値の比較を行った。しかし、ノイズ等の影響で多少のずれが生じる。よって、フィードバック入力により目標角度に追従するようにした。Fig. 2.15 はその実験結果である。シミュレーションと実験値との偏差があり、時間の経過とともにずれていくことが考えられる。こういった外乱による偏差を少なくし、目標角度に追従するように Fig. 2.14 のようにフィードバック系の構成を組む。

Fig. 2.14 は入力電圧を逆モータモデルにより目標軌道 θ_{ref} が生成される。そして、フィードバック系により偏差を取り PID コントローラで追従させる。このときのモータモデル、実験装置への入力電圧を u_{sim} , u_{exp} , そして、この時の出力を θ_{sim} , θ_{exp} とする。また u_{sim} , u_{exp} の偏差を u_{fb} として比較検証を行った。今回、すべてのフィードバックゲインは 0.03 とした。

その検証を行った結果を Fig. 2.15 に示す。1 段目のグラフは実線が目標角度で振幅 0.2[m], 周期 $2\pi/3$ の正弦波を逆モータモデルに入力し、その応答である。破線は実験装置のカウンタから得られた角度である。2 段目のグラフは目標角度が逆モータモデルに入力されたときの電圧出力の実験値である。3 段目には u_{fb} を表し、シミュレーションと実際に入力した電圧の偏差である。これらの結果から K_{m1} , T_{m1} は正しく求まったことがわかる。

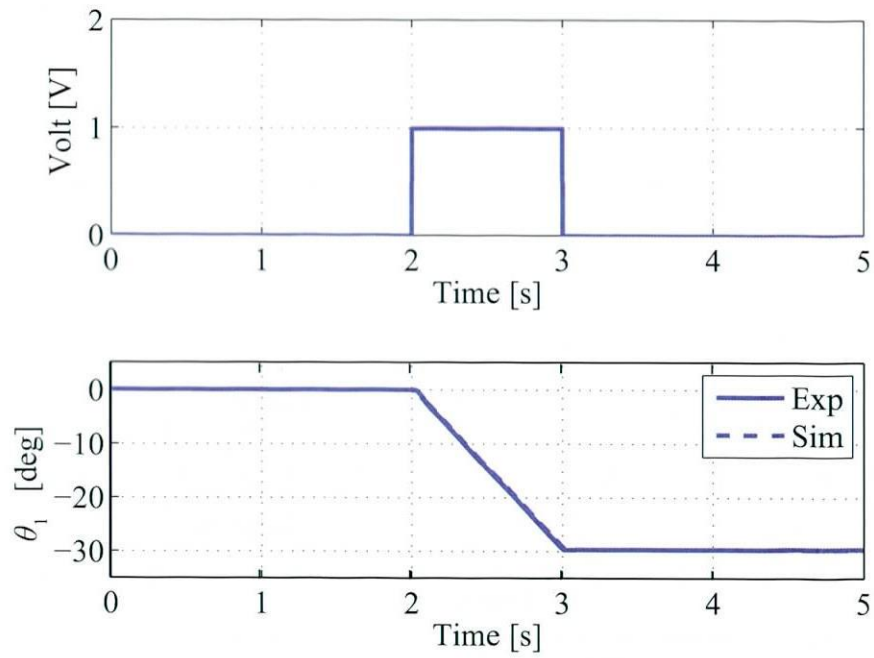


Fig. 2.13 Identify 1st motor of step input

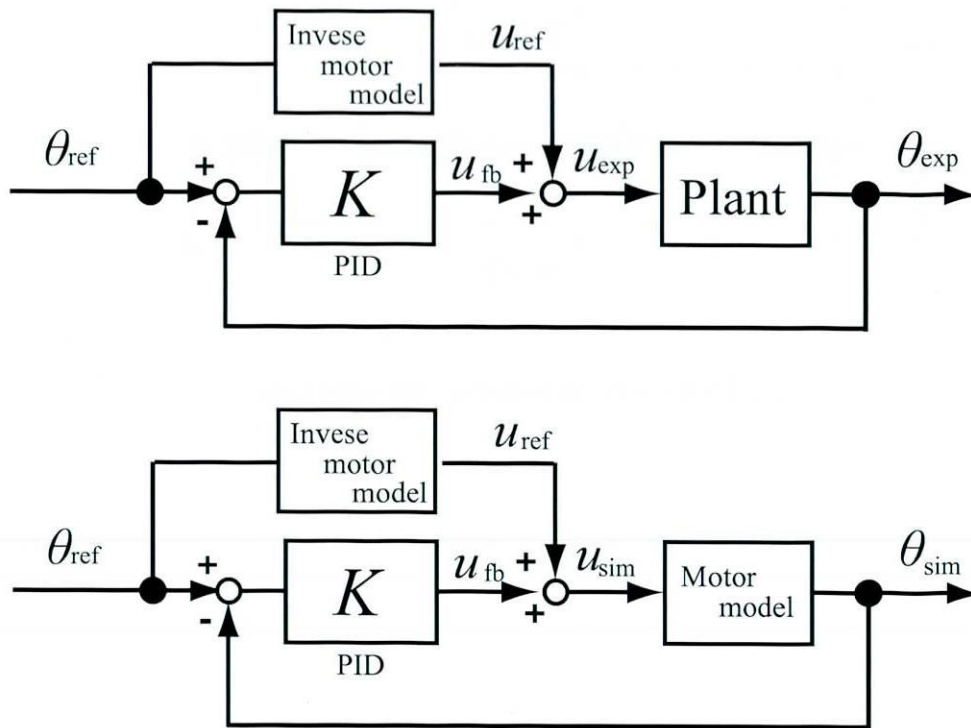


Fig. 2.14 Block diagram of feedback for motor control

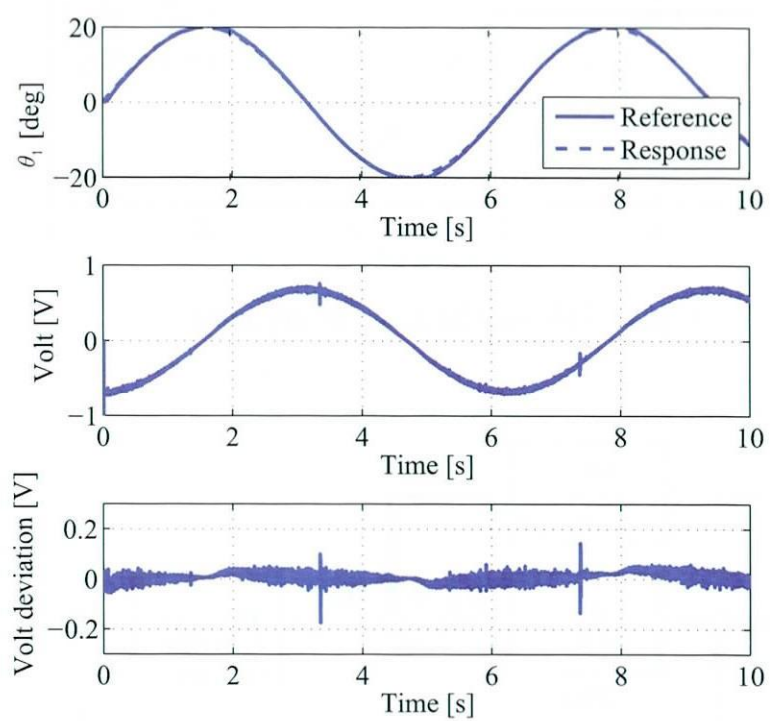


Fig. 2.15 Identify 1st motor of sine wave input

- 第2軸のモータ同定

第2軸のサーボモータも同様に, 入力電圧から回転角度までの伝達関数を1次遅れ系として Eq.(2.14) のように表現する. 第2軸モータにステップ入力で1[V]の電圧を1秒間与え, 実験値とシミュレーション値が一致するように K_{m2} , T_{m2} の値を試行錯誤により決定した. 破線がシミュレーションで実線が実験値の値である. Fig. 2.16 からシミュレーションと実験値が一致しているのがわかる.

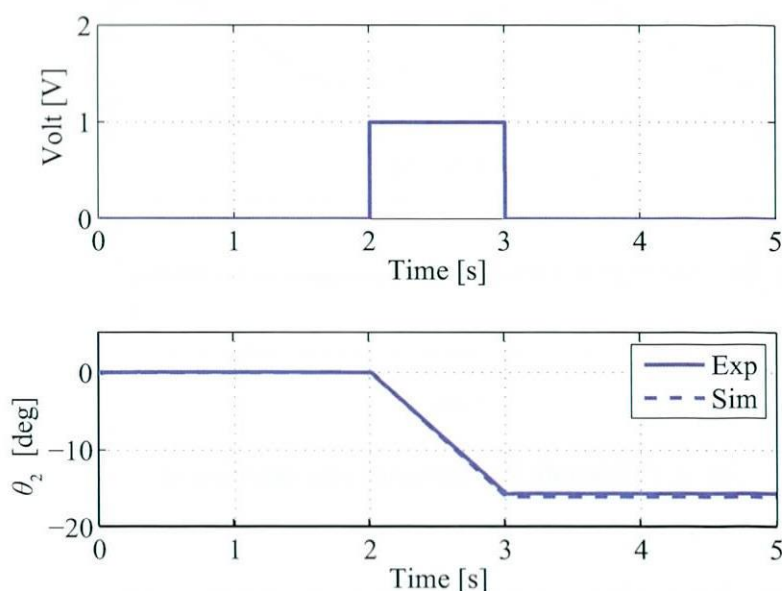


Fig. 2.16 Identify 2nd motor of step input

次に同定結果の検証を行った. 求まった K_{m2} , T_{m2} を用いて, 電圧1[V], 周期 $2\pi/3$ の sin 波を10秒間フィードフォワードで入力して, シミュレーションと実験値の比較を行った. その後ノイズ等の影響を考慮して, フィードバックによるPID制御を行った. Fig. 2.17 がその実験結果である. 1段目のグラフは実線が目標角度で振幅0.2[m], 周期 $2\pi/3$ の正弦波を逆モータモデルに入力し, その応答である. 破線は実験装置のカウンタから得られた角度である. 2段目のグラフは目標角度が逆モータモデルに入力されたときの電圧出力の実験値である. 3段目には u_{fb} を表し, シミュレーションと実際に入力した電圧の偏差である. これらの結果から K_{m2} , T_{m2} は正しく求まったことがわかる.

- 第3軸のモータ同定

第3軸のサーボモータも同様に, 入力電圧から回転角度までの伝達関数を1次遅れ系として Eq.(2.14) のように表現する. 実験値とシミュレーション値が一致するように K_{m3} , T_{m3} の値を試行錯誤により決定した結果, Fig. 2.18 からシミュレーションと実験値が一致しているのがわかる.

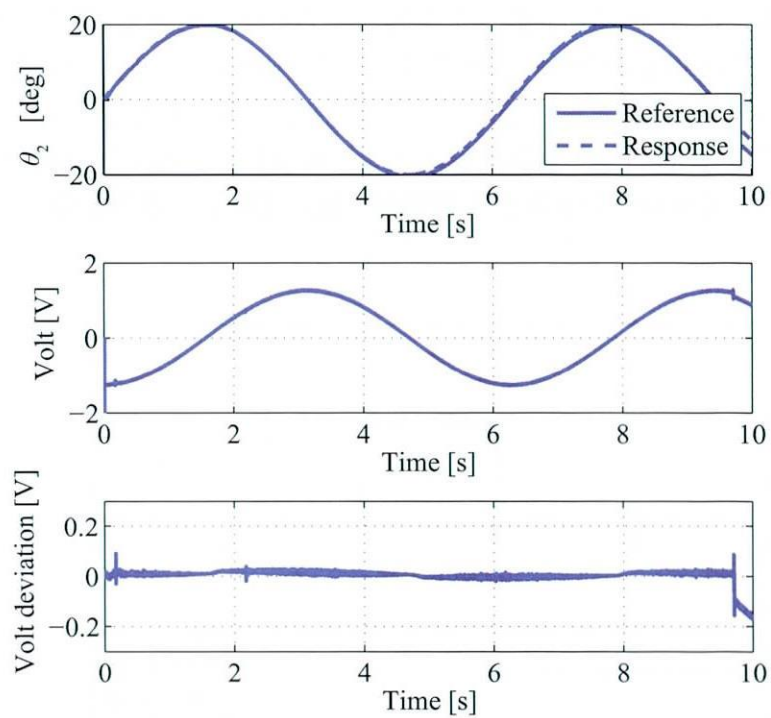


Fig. 2.17 Identify 2nd motor of sine wave input

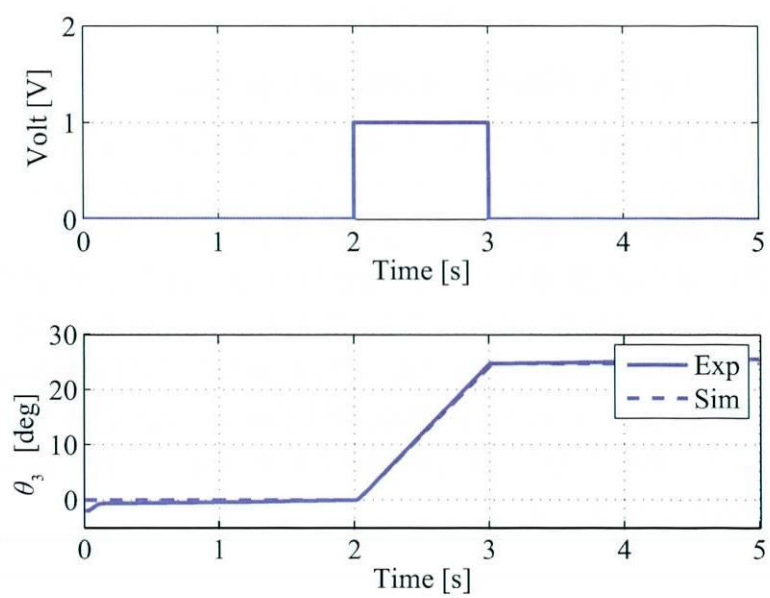


Fig. 2.18 Identify 3rd motor of step input

次に同定結果の検証を行った. 求めた K_{m3} , T_{m3} を用いて, 電圧 1[V], 周期 $2\pi/3$ の sin 波を 10 秒間フィードフォワードで入力して, シミュレーションと実験値の比較を行った. ノイズ等の影響を考慮して, フィードバックによる PID 制御を行った. Fig. 2.19 がその実験結果である. 1 段目のグラフは実線が目標角度で振幅 0.2[m], 周期 $2\pi/3$ の正弦波を逆モータモデルに入力し, その応答である. 破線は実験装置のカウンタから得られた角度である. 2 段目のグラフは目標角度が逆モータモデルに入力されたときの電圧出力の実験値である. 3 段目には u_b を表し, シミュレーションと実際に入力した電圧の偏差である. 電圧の偏差を見ても第 3 軸はオフセットが大きく, アンプの調整だけではできなかったことがわかる. これらの結果から K_{m3} , T_{m3} は正しく求めたことがわかる.

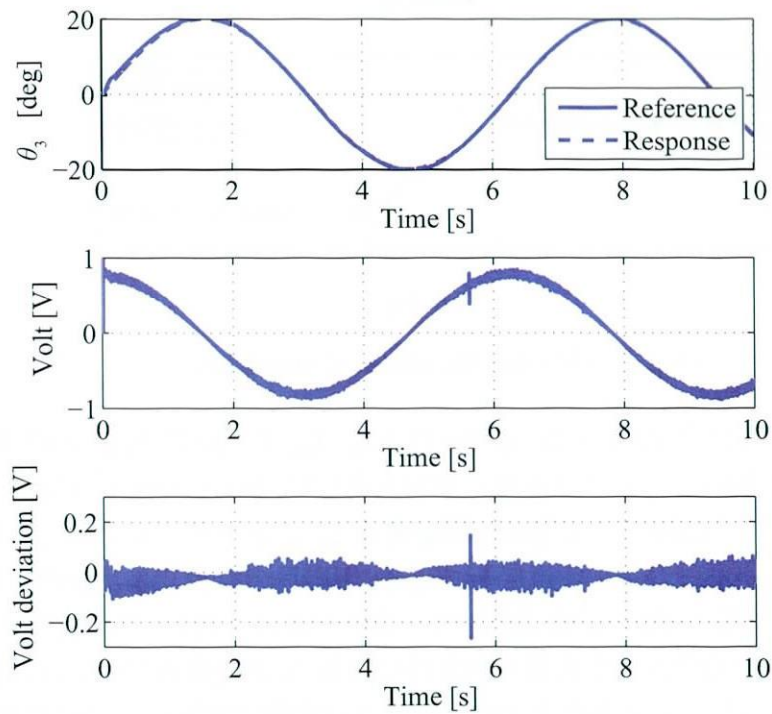


Fig. 2.19 Identify 3rd motor of sine wave input

- 第4軸のモータ同定

第4軸のサーボモータも同様に, 入力電圧から回転角度までの伝達関数を1次遅れ系として Eq.(2.14) のように表現する. 実験値とシミュレーション値が一致するように K_{m4} , T_{m4} の値を試行錯誤により決定した結果, Fig. 2.20 からシミュレーションと実験値が一致しているのがわかる.

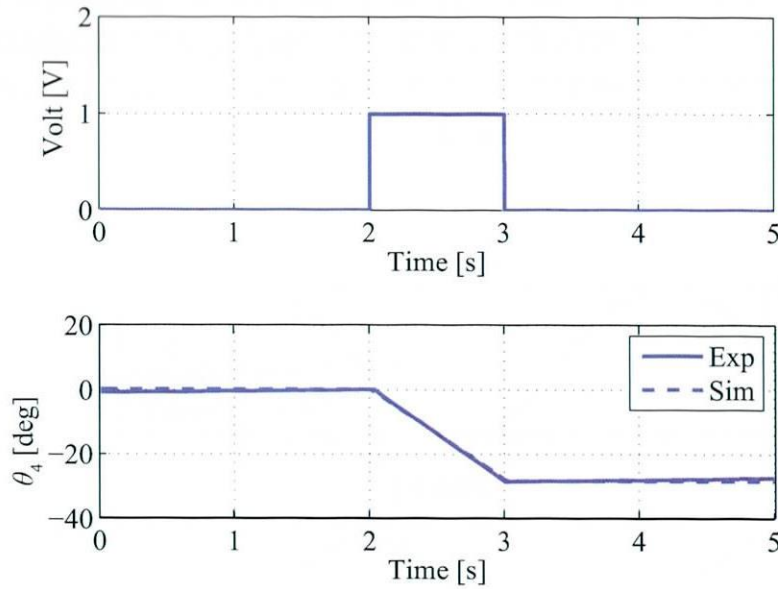


Fig. 2.20 Identify 4th motor of step input

次に同定結果の検証を行った. 求まった K_{m4} , T_{m4} を用いて, 電圧1[V], 周期 $2\pi/3$ の sin 波を10秒間フィードフォワードで入力して, シミュレーションと実験値の比較を行った. その後ノイズ等の影響を考慮して, フィードバックによるPID制御を行った. Fig. 2.21 がその実験結果である. 1段目のグラフは実線が目標角度で振幅0.2[m], 周期 $2\pi/3$ の正弦波を逆モータモデルに入力し, その応答である. 破線は実験装置のカウンタから得られた角度である. 2段目のグラフは目標角度が逆モータモデルに入力されたときの電圧出力の実験値である. 3段目には u_{fb} を表し, シミュレーションと実際に入力した電圧の偏差である. 第4軸も3軸と同じくオフセットが大きく, フィードバックによる制御が必要である. これらの結果から K_{m4} , T_{m4} は正しく求まったことがわかる.

- 第5軸のモータ同定

第5軸のサーボモータも同様に, 入力電圧から回転角度までの伝達関数を1次遅れ系として Eq.(2.14) のように表現する. 実験値とシミュレーション値が一致するように K_{m5} , T_{m5} の値を試行錯誤により決定した結果, Fig. 2.22 からシミュレーションと実験値が一致しているのがわかる.

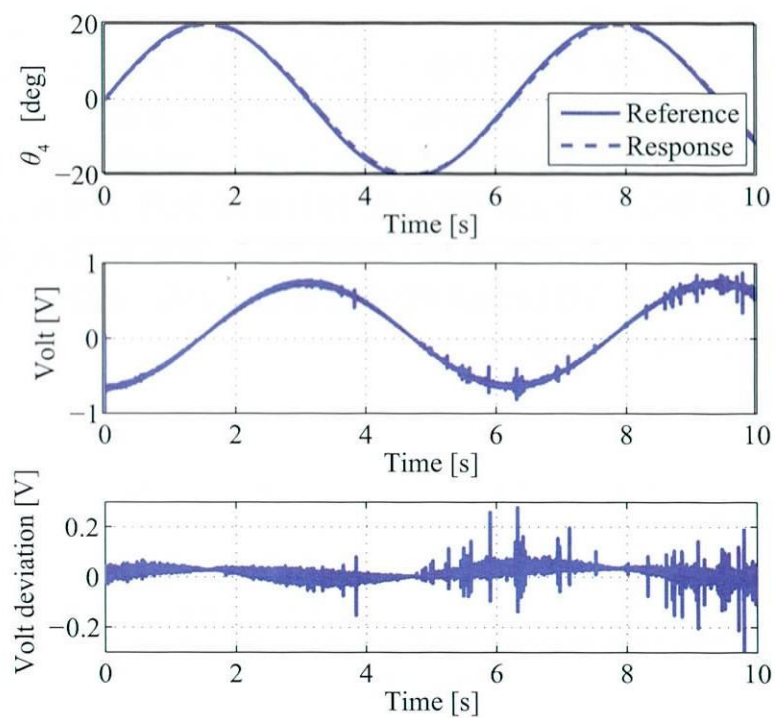


Fig. 2.21 Identify 4th motor of sine wave input

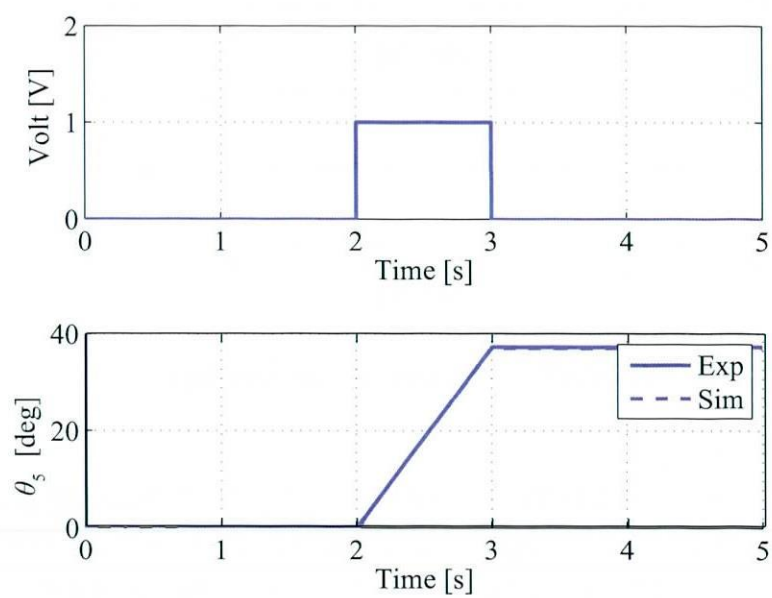


Fig. 2.22 Identify 5th motor of step input

次に同定結果の検証を行った。求まった K_{m5} , T_{m5} を用いて、電圧1[V], 周期 $2\pi/3$ の \sin 波を10秒間フィードフォワードで入力して、シミュレーションと実験値の比較を行った。その後ノイズ等の影響を考慮して、フィードバックによるPID制御を行った。Fig. 2.23がその実験結果である。1段目のグラフは実線が目標角度で振幅0.2[m], 周期 $2\pi/3$ の正弦波を逆モータモデルに入力し、その応答である。破線は実験装置のカウンタから得られた角度である。2段目のグラフは目標角度が逆モータモデルに入力されたときの電圧出力の実験値である。3段目には u_{fb} を表し、シミュレーションと実際に入力した電圧の偏差である。これらの結果から K_{m5} , T_{m5} は正しく求まったことがわかる。

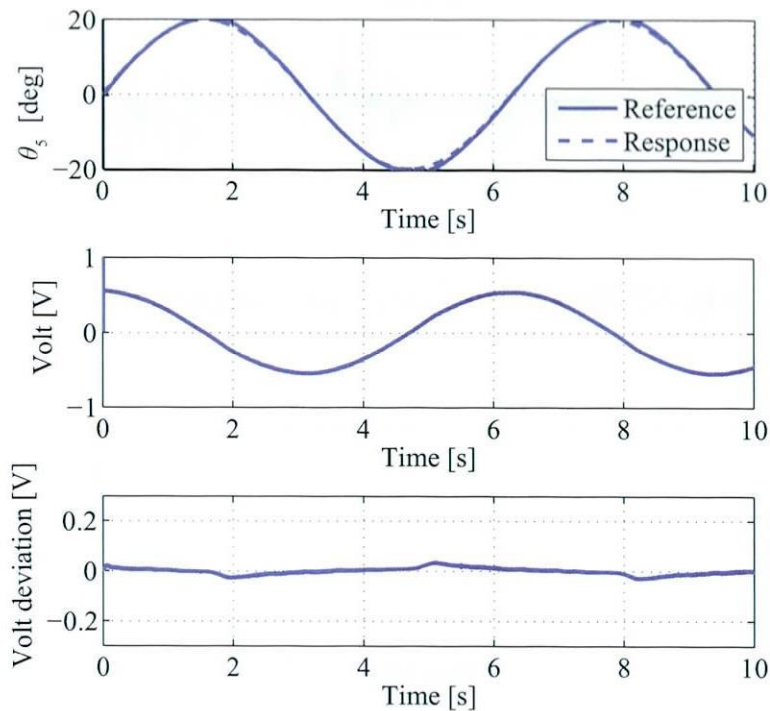


Fig. 2.23 Identify 5th motor of sine wave input

- 第6軸のモータ同定

第6軸のサーボモータも同様に、入力電圧から回転角度までの伝達関数を1次遅れ系として Eq.(2.14) のように表現する。実験値とシミュレーション値が一致するように K_{m6} , T_{m6} の値を試行錯誤により決定した結果、Fig. 2.24 からシミュレーションと実験値が一致しているのがわかる。

次に同定結果の検証を行った。求まった K_{m6} , T_{m6} を用いて、電圧1[V], 周期 $2\pi/3$ の \sin 波を10秒間フィードフォワードで入力して、シミュレーションと実験値の比較を行った。その後ノイズ等の影響を考慮して、フィードバックによるPID制御を行った。Fig. 2.25がその実験結果である。1段目のグラフは実線が目標角度で振幅

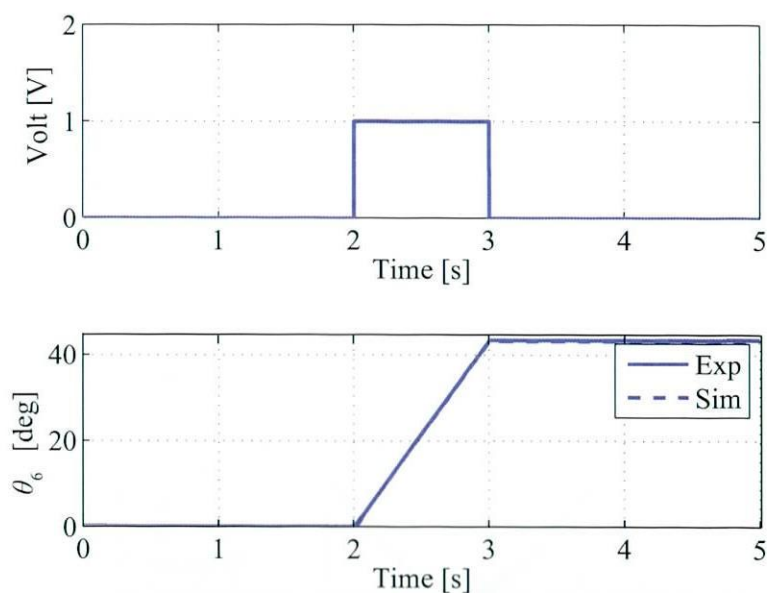


Fig. 2.24 Identify 6th motor of step input

Table 2.5 Parameter of each motion motor

	Motor Gain K_m [deg/(s·V)]	Time Constant T_m [s]
θ_1 motor	-29.650	0.020
θ_2 motor	-15.520	0.025
θ_3 motor	24.870	0.015
θ_4 motor	-29.550	0.010
θ_5 motor	37.600	0.005
θ_6 motor	44.940	0.005

0.2[m], 周期 $2\pi/3$ の正弦波を逆モータモデルに入力し, その応答である. 破線は実験装置のカウンタから得られた角度である. 2 段目のグラフは目標角度が逆モータモデルに入力されたときの電圧出力の実験値である. 3 段目には u_{fb} を表し, シミュレーションと実際に入力した電圧の偏差である. これらの結果から K_{m6} , T_{m6} は正しく求まったことがわかる.

そして, Table 2.5 には各ゲインと時定数をまとめたものを表す.

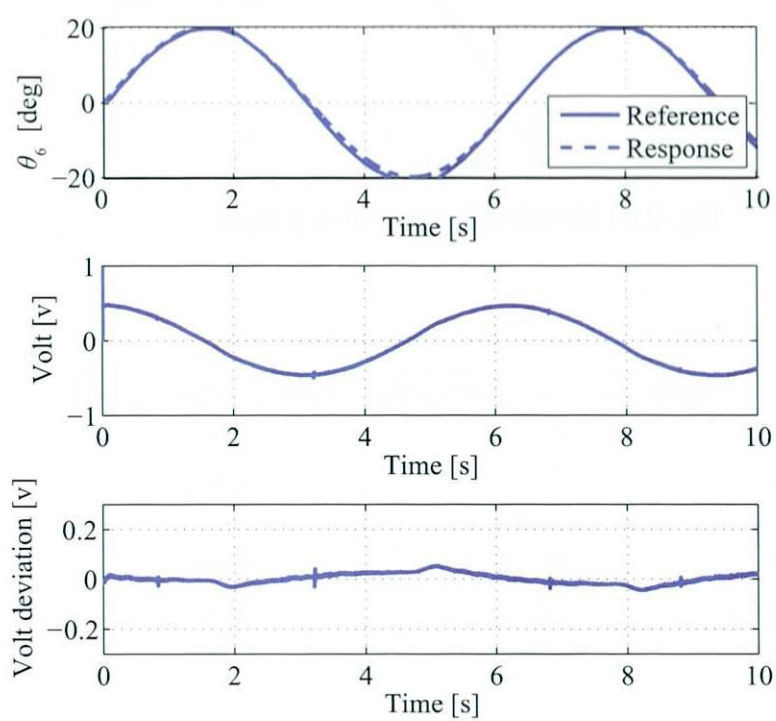


Fig. 2.25 Identify 6th motor of sine wave input

2.5 溢流の防止を考慮した搬送制御系設計

2.5.1 食事支援ロボットによる液状食物の搬送

食事支援ロボットによる液状食物の搬送を行う際、食事支援の動作を“液体を掬う”と“口元まで運ぶ”という2つの動作に分け、食べ物を掬うと同時に、Fig. 2.26に示すように利用者に対してスプーンが常に同じ方向、つまりFig. 2.3における Y_s 軸方向にのみスプーンを搬送する。液体の挙動の特徴として、上下方向の搬送では液体は揺れず^[28]、そのため上下方向のスプーンの移動では液体を溢すことはないことから、本研究で対象とする3次元空間での液体搬送問題は、単純な液体の直線搬送制御問題と等しく扱う。加えて、一定速度での搬送では液体は揺れないことから、加減速度カーブが液体の溢流抑制に重要な要素となる。

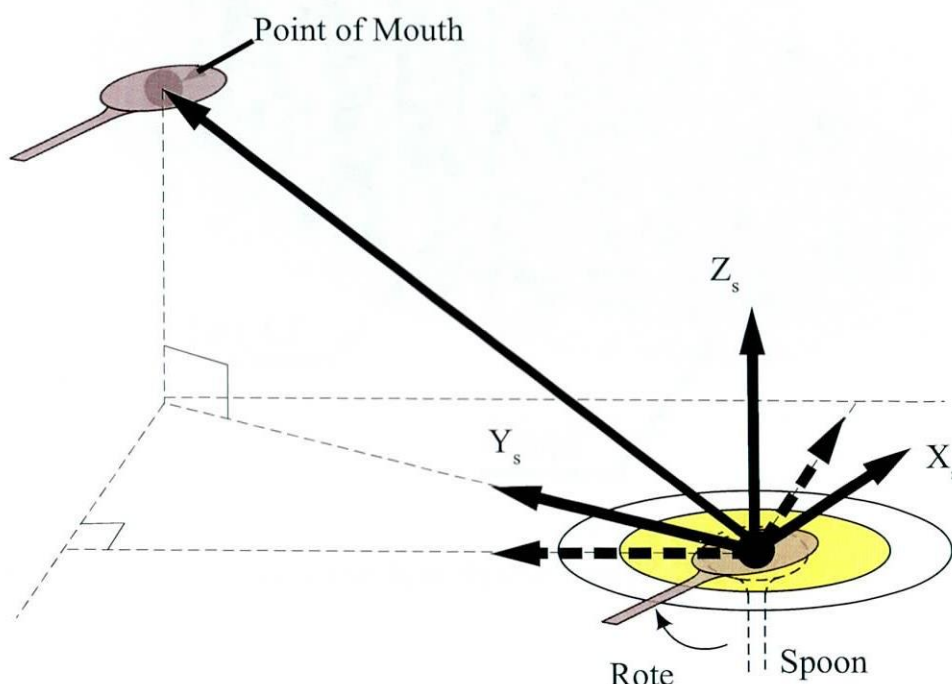


Fig. 2.26 Direction of the spoon transfer

一方、液体の固有振動数付近でコントローラのゲインを下げ、液面振動を抑制していた従来までの液体搬送制御手法^[28]では、振動抑制は可能であるものの、表面張力の影響や溢れるという現象を制御することは難しい。さらには、これらの現象を実時間で計測し、フィードバック制御することも困難である。

そこで本研究では、液体の溢流抑制は加減速度カーブによる影響が大きいことに着目し、その加速度カーブの最適化を行う。具体的には、流体解析シミュレータによる溢流現象を同定し、高次のスプライン関数で作られる加速度カーブの未知パラメータを、液体の溢流を評価に入れた制約条件下の基で最適化を行う。また減速度カーブは

スプーンが X_s 軸に対して線対称であることから、加速度カーブを反転した物を用いることにする。

溢流を制約条件として追加するために、シミュレータで液体の溢流を判定する必要がある。そこで流体解析シミュレータにおけるスプーンの縁に、Fig. 2.27 に示すような測定面を設定し、この測定面を液体が通り過ぎた場合を溢流とした。

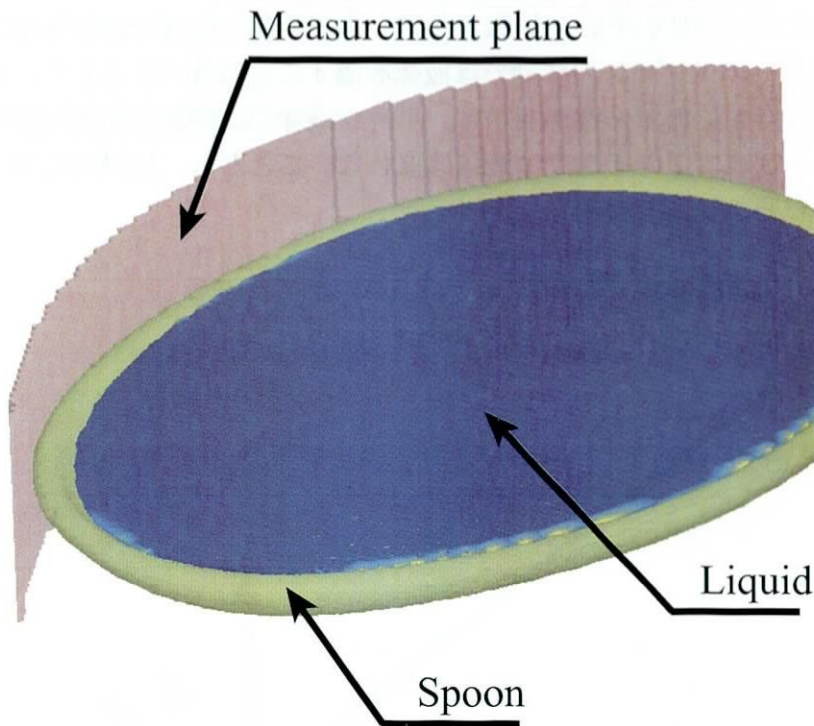


Fig. 2.27 Measurement plane of the spoon

2.5.2 加速カーブの定式化

搬送軌道は速度カーブを Fig. 2.28 に示す様に加速区間、定速区間、減速区間の3つに区分する。また、その加速区間、減速区間である速度カーブは高次のスプライン関数を用いて生成する。以下では加速区間に対する速度カーブの式の導出を行うことにする。

加速度の急激な変化による溢流を防止するため、本研究では、加速カーブを Eq.(2.16) を微分した Eq.(2.17) に示す関数を用いて表現する。ここで、 $v(t)$ は速度、 $a(t)$ は加速度、 t は時間、 a_i ($i = 0 \sim n$) は定数であり、また加加速度を滑らかにするための最低限の次数として、7 次関数のスプラインとした ($n = 7$) を選択した。

$$v(t) = \sum_{i=0}^n a_i t^i \quad (2.16)$$

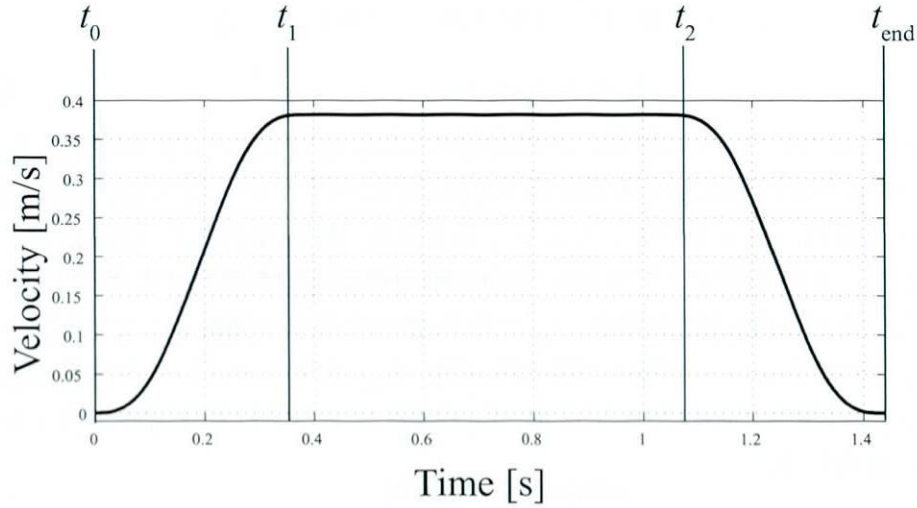


Fig. 2.28 Planning of velocity curve to derive the transfer path with spilling avoidance

$$a(t) = \sum_{i=0}^{n-1} (a_{i+1} t^i)(i+1) \quad (2.17)$$

また搬送の初期条件および加速終了時間 t_1 における終端条件を考慮して、 $v(0) = 0.0[\text{m/s}]$, $a(0) = 0.0[\text{m/s}^2]$, $a'(0) = \sum_{i=0}^{n-2} (a_{i+2} t^i)(i^2 + 3i + 2) = 0.0[\text{m/s}^3]$, $v(t_1) = V_1[\text{m/s}]$, $a(t_1) = 0.0[\text{m/s}^2]$, $a'(t_1) = 0.0[\text{m/s}^3]$ とする。ここで $a'(t)$ は加速度 $a(t)$ の微分である。

結果として、これら条件から Eq.(2.18)~Eq.(2.21) を得ることができる。なお、これらの導出に至っては付録 B に示しておく。

$$a_0 = a_1 = a_2 = 0 \quad (2.18)$$

$$a_5 = \frac{6V_1 - 6a_7 t_1^7 - 3a_6 t_1^6}{t_1^5} \quad (2.19)$$

$$a_4 = \frac{-15V_1 + 8a_7 t_1^7 + 3a_6 t_1^6}{t_1^4} \quad (2.20)$$

$$a_3 = \frac{10V_1 - 3a_7 t_1^7 - a_6 t_1^6}{t_1^3} \quad (2.21)$$

探索する未知パラメータは t_1, V_1, a_7, a_6 の4つとなる。ここで V_1 は、あらかじめモーションキャプチャを用いて一般成人男性が食事をする際の動作を計測し、物を掬ってから口に持ってくるまでの平均的な速度を解析することにより求めた。その結果、 $V_1 = 0.37[\text{m/s}]$ とした。

パラメータ最適化のために、本問題を Eq.(2.22) に示すようにペナルティ項を用いた制約付きの最適化問題に定式化する。

$$J_s = T_s + J_p \quad (2.22)$$

$$T_s = \min \{t \mid |P_f - P(t + \sigma)| < 0.001, \sigma \geq 0\} \quad (2.23)$$

$$J_p = \omega_1 + \omega_2 + \omega_3 + \dots \quad (2.24)$$

ここで Eq.(2.23) について, t は時間, P は時刻 t における位置, σ は正の整数, T_s は整定時間, P_f は加速終了時点の位置である. Eq.(2.24) のペナルティ項 J_p は, 制約条件を満たさなかった場合, $\omega_k = 10^8 (k = 1, 2, \dots)$ を与える. 制約条件には, 速度が正となるための Eq.(2.25), 6 自由度マニピュレータの各軸の速度制約を満たすための Eq.(2.26), 液体を溢さないための Eq.(2.27) を用い, これらをペナルティ関数として表現した. ここで $\max|\dot{\theta}_j|$ はマニピュレータ各軸の最大角速度, $\dot{\theta}_{j\max}$ はマニピュレータ各軸の最大角速度制約, Q_{spill} は液体が溢れた量を示す. Eq.(2.27) に至っては流体解析シミュレータを援用して評価を行う.

$$\min[v(t)] < 0 \text{ [m/s]} \quad (2.25)$$

$$\max|\dot{\theta}_j| > \dot{\theta}_{j\max} \text{ [rad/s]} (j = 1 \sim 6) \quad (2.26)$$

$$Q_{\text{spill}} > 0 \text{ [ℓ]} \quad (2.27)$$

2.5.3 遺伝的アルゴリズム

遺伝的アルゴリズム (GA) は, 特に広く普及し, 多くの最適化アルゴリズムの基礎となっている. 遺伝的アルゴリズムは自然界における生物集団の遺伝と進化の過程における自然淘汰, 突然変異をモデル化したアルゴリズムであり, 確率的な最適化手法に分類される. 遺伝的アルゴリズムは厳密に確立された一つのアルゴリズムではないため, 柔軟性に富み, 非常に広範囲の問題に適応することが可能である.

遺伝的アルゴリズムにおける各個体の適応度に応じて次世代の個体集団に残すべき個体を選ぶことを選択 (Selection) という. 選択法は, ランキング選択, トーナメント選択等が考案されている. 本研究では, 適応度に応じた確率で次世代に残す個体を決定するルーレット選択である. また最良の個体を次世代に残すエリート保存 (Elite preservation) を適用することにより, 一度現れた優れた個体が交叉や突然変異などにより消滅してしまうことを防ぎ, より効率的な最適解探索を行っている.

また2つの個体を組み合わせて新たな個体を作ること交叉 (Crossover) という. 交叉をするにあたり, 各個体を2進数に置き換え遺伝子に見立てる. 多変数の場合には横に並べることにより1つの個体を作成する (Coding). 交叉法についても様々な方法が提案されているが, 本研究で比較する遺伝的アルゴリズムでは一点交叉 (One-point crossover) を用いる. 一点交叉とは選択された2つの個体 (親) から, ランダムに交叉位置 (Crossover point) を決定し, 親個体の遺伝子を切断し掛け合わせる方法である. 比較する遺伝的アルゴリズムでは適応度が上位の個体から, 一定の確率でそれよりも適応度が低い個体を交叉させる. 切断した遺伝子を掛け合わせるため, 1組の親個体から2つの個体を作成されることになるが, 個体集団の多様性を維持するため, どちらか1つのみを選択する. 遺伝的アルゴリズムの概要を Fig. 2.29 に, 交叉の概要を Fig. 2.30 に示す.

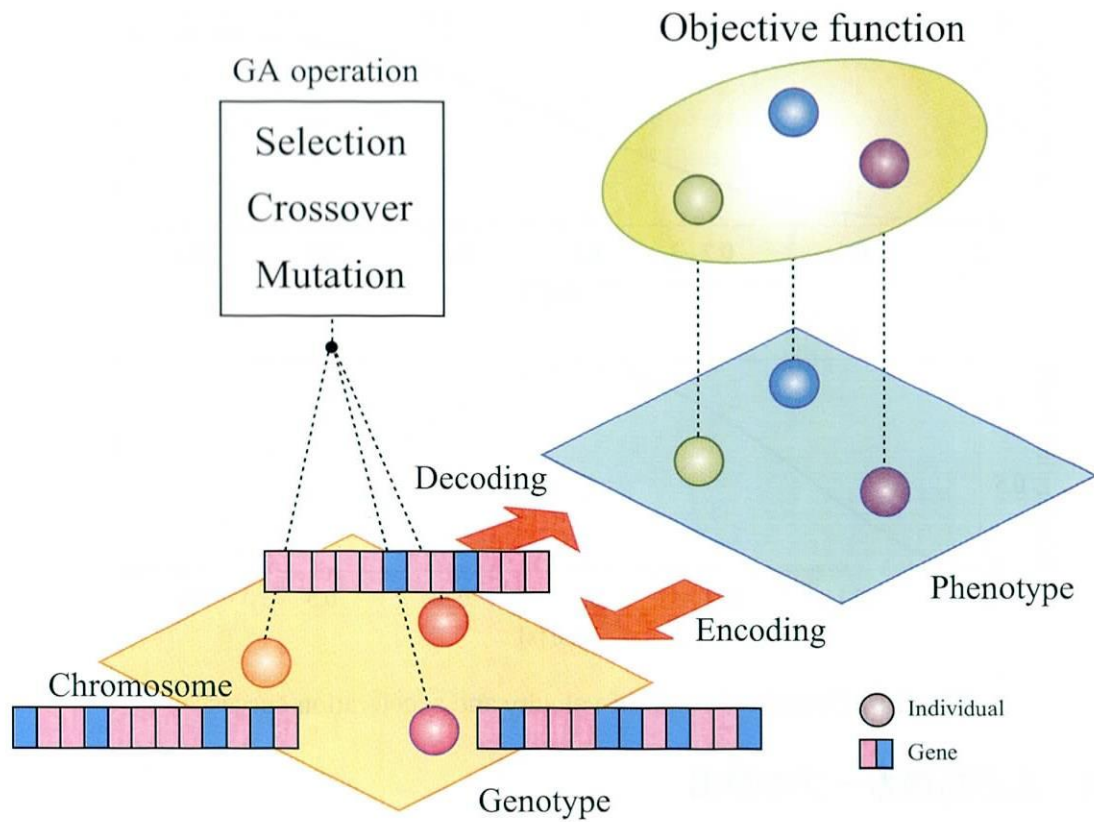


Fig. 2.29 Conceptual diagram of Genetic Algorithm

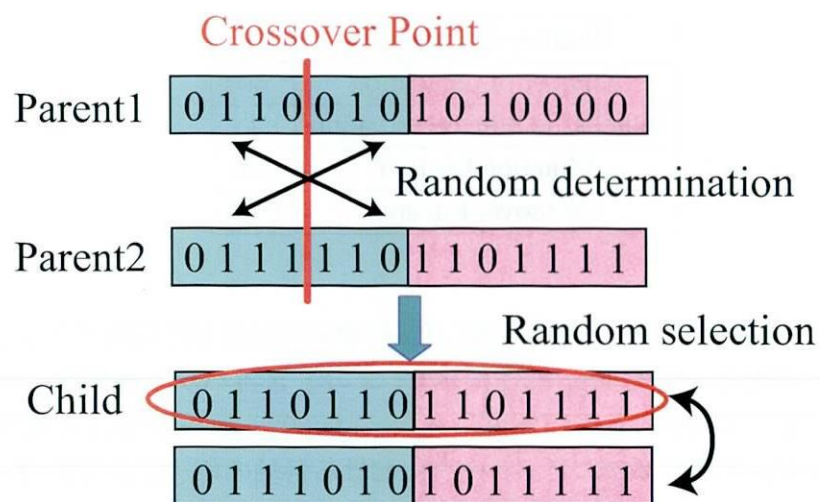


Fig. 2.30 Example of one-point crossover

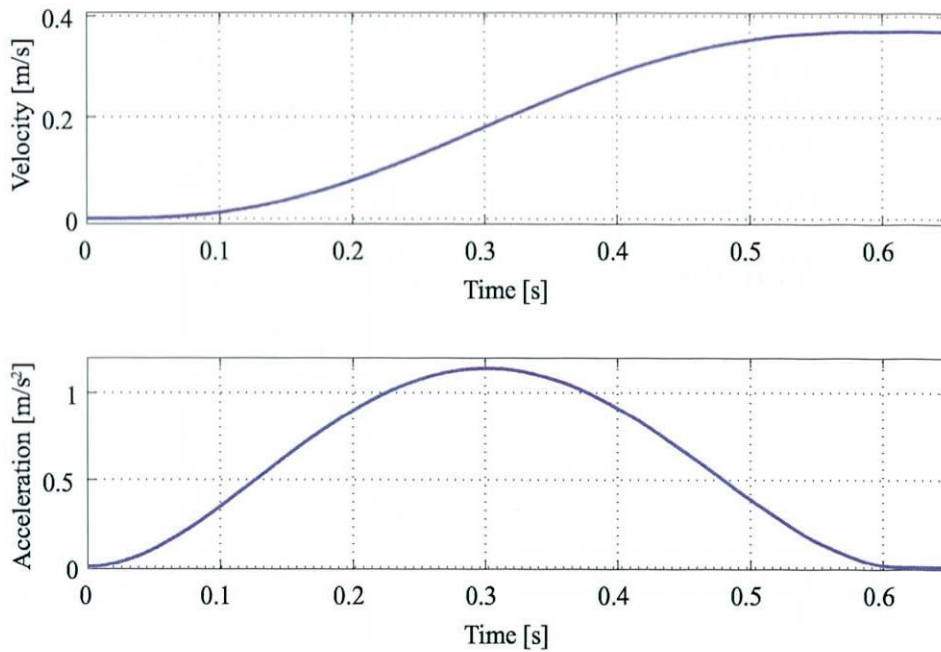


Fig. 2.31 Optimized result of velocity and acceleration curves

2.5.4 最適加速カーブの導出

マニピュレータの根元に原点を置き, $(0.45, 0.00, 0.30)$ [m] の位置を始点, Y_o 軸方向に 0.30 [m] 移動した $(0.45, 0.30, 0.30)$ [m] の位置を終点と設定し, 最適化問題を解いた. このとき, 遺伝的アルゴリズムのパラメータは Table 2.6 のように設定した.

Table 2.6 Parameters for genetic algorithm

Number of variable	3
Number of population	30
Number of elite preservation	1
Mutation fraction	0.02
Crossover fraction	0.80

結果として, 13 世代目で 1.42 [s] に収束した. 計算時間は約 100 時間であった. 求まった速度カーブと加速カーブを Fig. 2.31 に示す. このとき各パラメータは $t_1 = 0.62$ [s], $a_7 = -14.808$, $a_6 = 25.073$, $a_5 = 12.407$, $a_4 = -37.692$, $a_3 = 16.365$ となった.

また Fig. 2.32 に, 6 自由度マニピュレータにおける各軸の角速度を示す. Table 2.1 に示す最大角速度制約と比較すると, 6 自由度マニピュレータの制約の範囲内で搬送を行っていることがわかる. さらに, Fig. 2.33 はシミュレーション結果を示しており, 液体を溢すことなく液体搬送していることが確認できる.

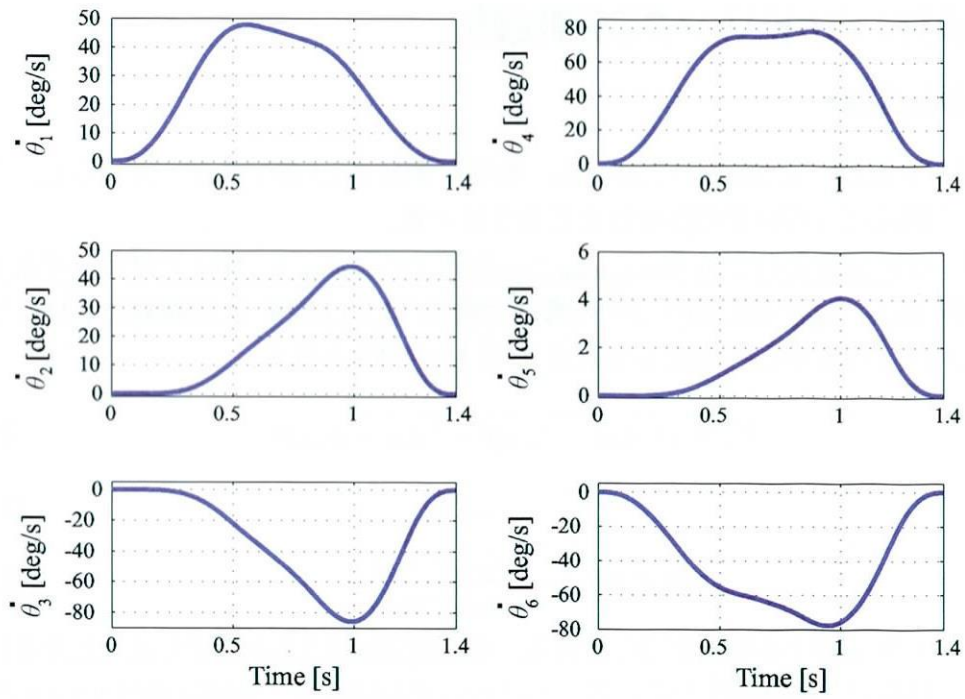


Fig. 2.32 Angular velocity of 6DOF manipulator

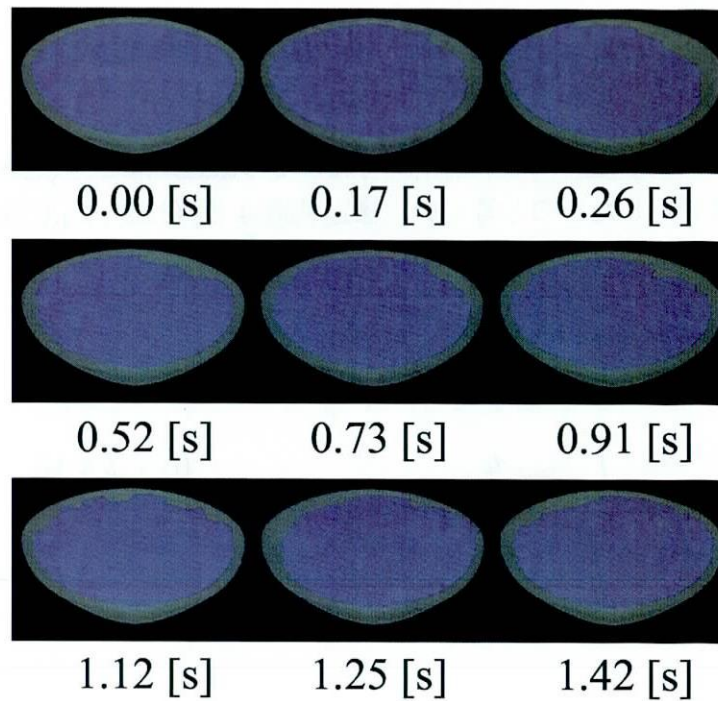


Fig. 2.33 Fluid analysis simulation result of liquid transfer

2.6 液体ハンドリング制御実験

2.6.1 軌道設計

3次元搬送軌道の設計について述べる。搬送軌道は2.5.1節でも述べたように、利用者に対して常に同じ方向で搬送を行えるようにする。

始点および終点をそれぞれ $(x_{\text{start}}, y_{\text{start}}, z_{\text{start}})$ [m], $(x_{\text{end}}, y_{\text{end}}, z_{\text{end}})$ [m] とすると平面方向の搬送距離 H_d は Eq.(2.28), 高さ方向の搬送距離 Z_d は Eq.(2.29), 初期座標における Y 軸を搬送方向に Z 軸を中心に回転した角度 θ_d は Eq.(2.30) となる。

$$H_d = \sqrt{(x_{\text{end}} - x_{\text{start}})^2 + (y_{\text{end}} - y_{\text{start}})^2} \quad (2.28)$$

$$Z_d = z_{\text{end}} - z_{\text{start}} \quad (2.29)$$

$$\theta_d = \arctan \left(\frac{x_{\text{end}} - x_{\text{start}}}{y_{\text{end}} - y_{\text{start}}} \right) \quad (2.30)$$

次に平面方向における搬送軌道を求める。搬送軌道は2.5.2小節でも述べたように加速区間、定速区間、減速区間からなる。したがって加速区間、減速区間は2.5.4小節で求めた、溢流抑制加速度カーブを用いて設計する。

定速区間の時間を求める必要がある。求めた速度カーブの関数を $f(t)$ とすると定速区間の時間は Eq.(2.31) となる。

$$t_2 - t_1 = \frac{\left(H_d - 2 \int_0^{t_1} f(t) dt \right)}{V_{\text{target}}} \quad (2.31)$$

ここで $2 \int_0^{t_1} f(t) dt$ は、初期速度から加速終了時間までの搬送距離と、減速区間から終了時間までの搬送距離を足したものと等しい。搬送軌道は Eq.(2.32), Eq.(2.33), Eq.(2.34) のようになる。

$$\int_0^t f(t) dt = \int_0^t \int_0^t \sum_{i=0}^{n-1} (a_{i+1} t^i) (i+1) dt^2 \quad (t_0 \leq t \leq t_1) \quad (2.32)$$

$$\int_0^t f(t) dt = \int_0^t V_{\text{target}} dt \quad (t_1 < t < t_2) \quad (2.33)$$

$$\int_0^t f(t) dt = - \int_0^t \int_0^t \sum_{i=0}^{n-1} (a_{i+1} t^i) (i+1) dt^2 \quad (t_2 \leq t \leq t_{\text{end}}) \quad (2.34)$$

2.6.2 3次元搬送

提案手法の有効性を示すために、比例制御による位置決め制御系、ハイブリッド整形法による振動制御系^[85]、溢流の防止を考慮した制御系による液体搬送制御実験を行った。ここで、搬送条件は $(0.51, 0.00, 0.21)$ [m] の位置を始点、始点から X_0 軸方向に 0.07 [m]、 Y_0 軸方向に 0.24 [m] そして Z_0 軸方向に 0.24 [m] 移動した $(0.58, 0.24, 0.45)$ [m] の位置を終点と設定とした。

比例制御系では、提案手法と搬送時間が等しくなるように比例ゲインを $K_p = 6.1$ と設定した。制御対象の固有振動数を特定するため、FFT 解析を用いて固有振動数を求めた結果、 3.90 [Hz] (24.49 [rad/s]) であった。この固有振動数を用い、ハイブリッド整形法により振動抑制コントローラを設計した。Fig. 2.34 ~ Fig. 2.36 に振動抑制コントローラを用いたときの6自由度マニピュレータの各軸の速度結果を示す。

各制御手法による制御実験結果の搬送軌道、速度カーブおよび加速度カーブを Fig. 2.37 に、また実際のスプーン上の CFD を Fig. 2.38 ~ Fig. 2.40 に示す。ここで、Fig. 2.37 の実線、破線、一点鎖線はそれぞれ溢流抑制制御、ハイブリッド整形法、そして比例制御による搬送軌道を示している。

Fig. 2.38 と Fig. 2.40 に示す比例制御と溢流抑制制御の実験結果から、搬送時間が等しく、軌道や速度および加速度が滑らかであるのにも関わらず、比例制御では搬送途中で液体を溢しているのがわかる。また Fig. 2.39 と Fig. 2.40 のハイブリッド整形法と溢流抑制制御の実験結果から、振動制御を行っても液体が溢れることが実際に確認できた。

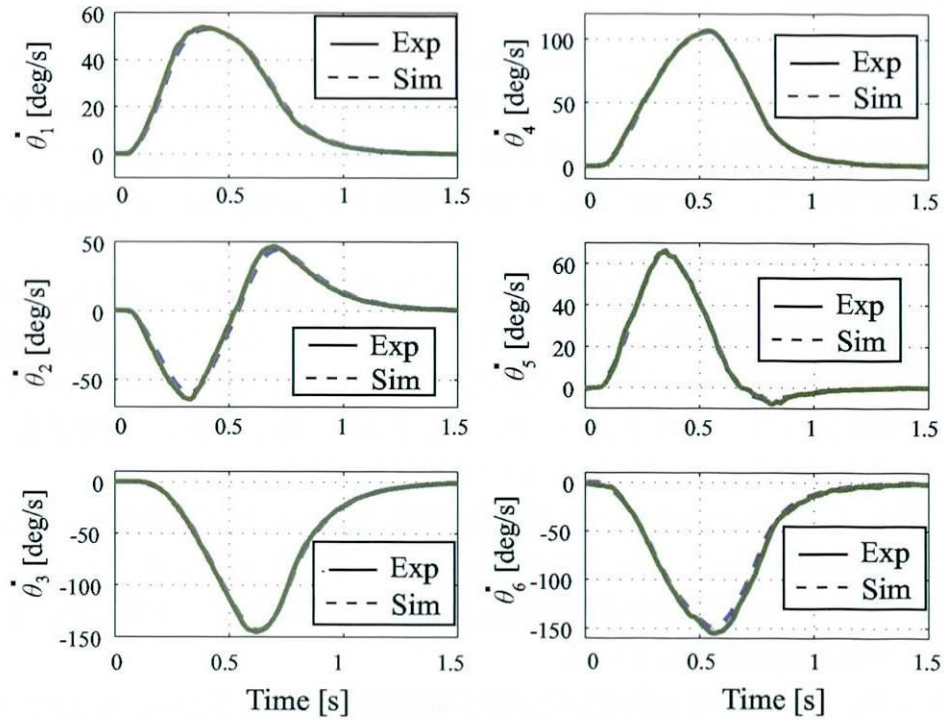


Fig. 2.34 Experimental results of each angular velocity of 6DOF manipulator by the proportional control

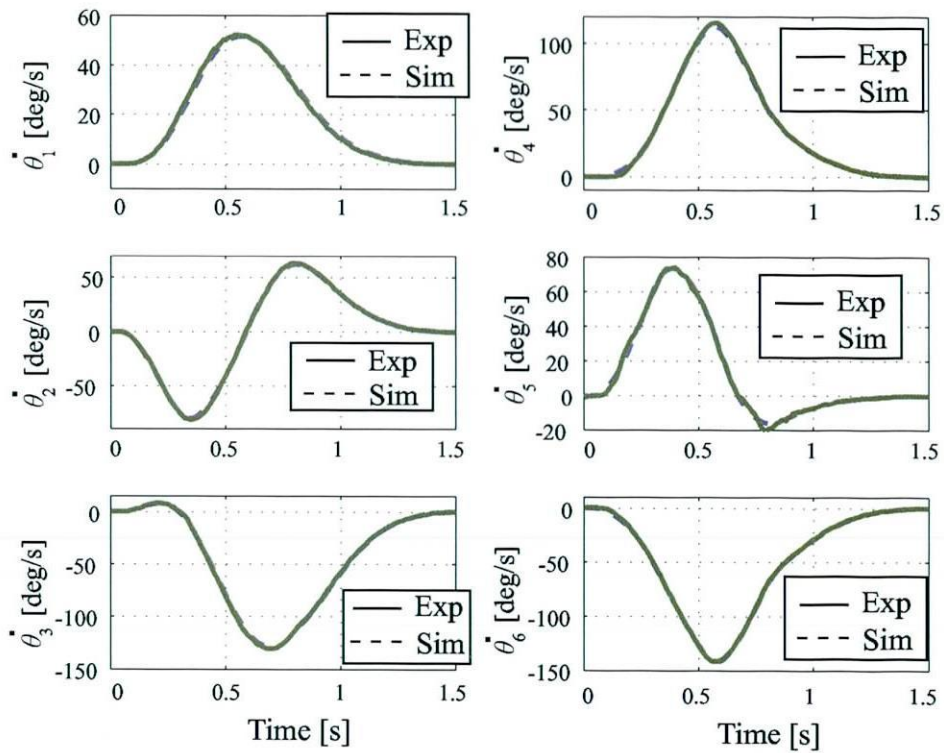


Fig. 2.35 Experimental results of each angular velocity of 6DOF manipulator by the hybrid shape approach

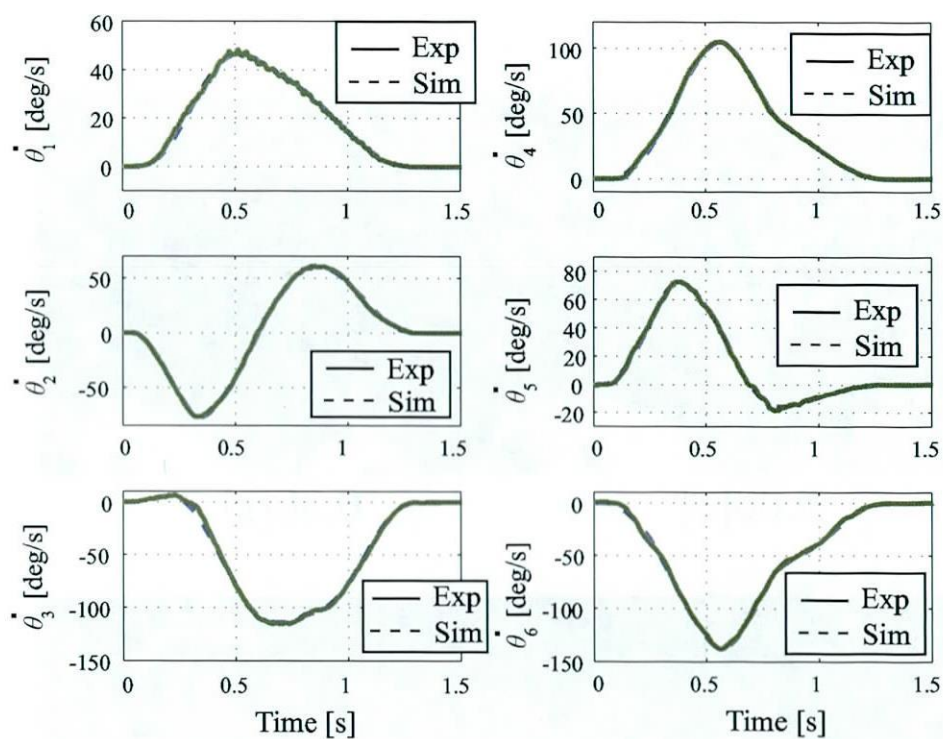


Fig. 2.36 Experimental results of each angle velocity of 6DOF manipulator by the proposed method

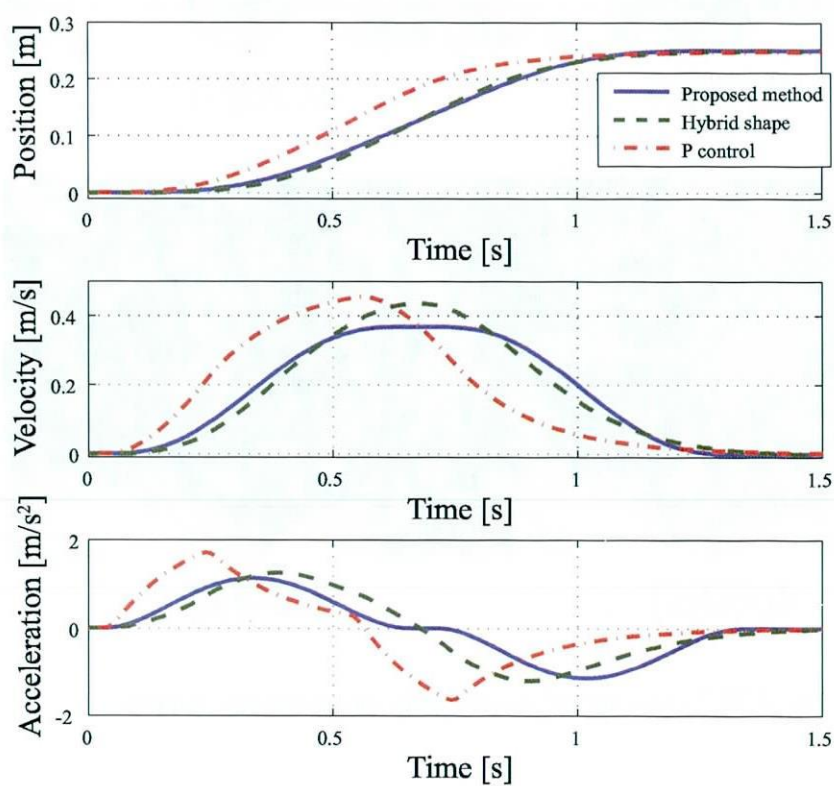


Fig. 2.37 Transfer path by the each controller

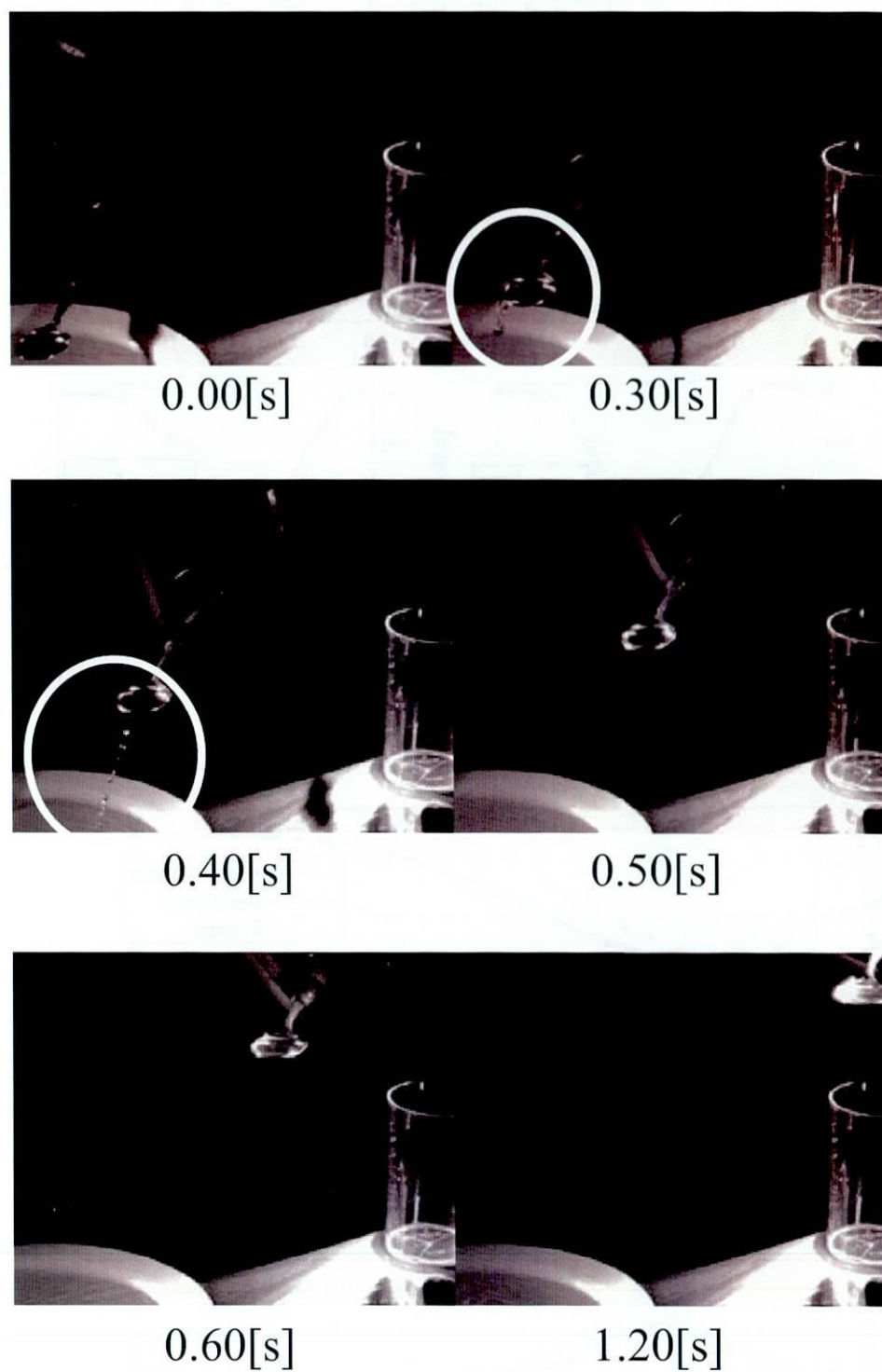


Fig. 2.38 Experimental result by the proportional control

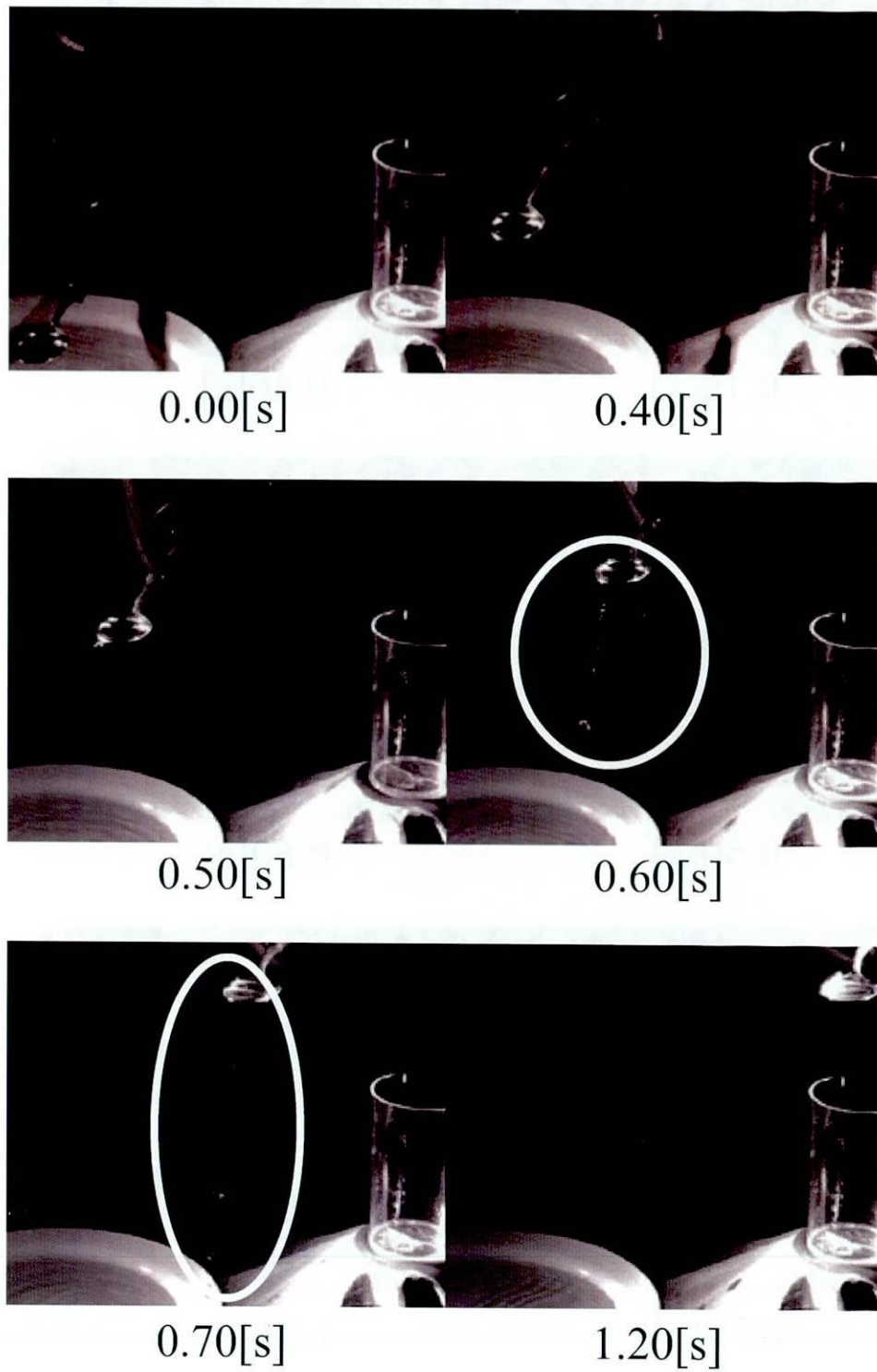


Fig. 2.39 Experimental result by the hybrid shape approach

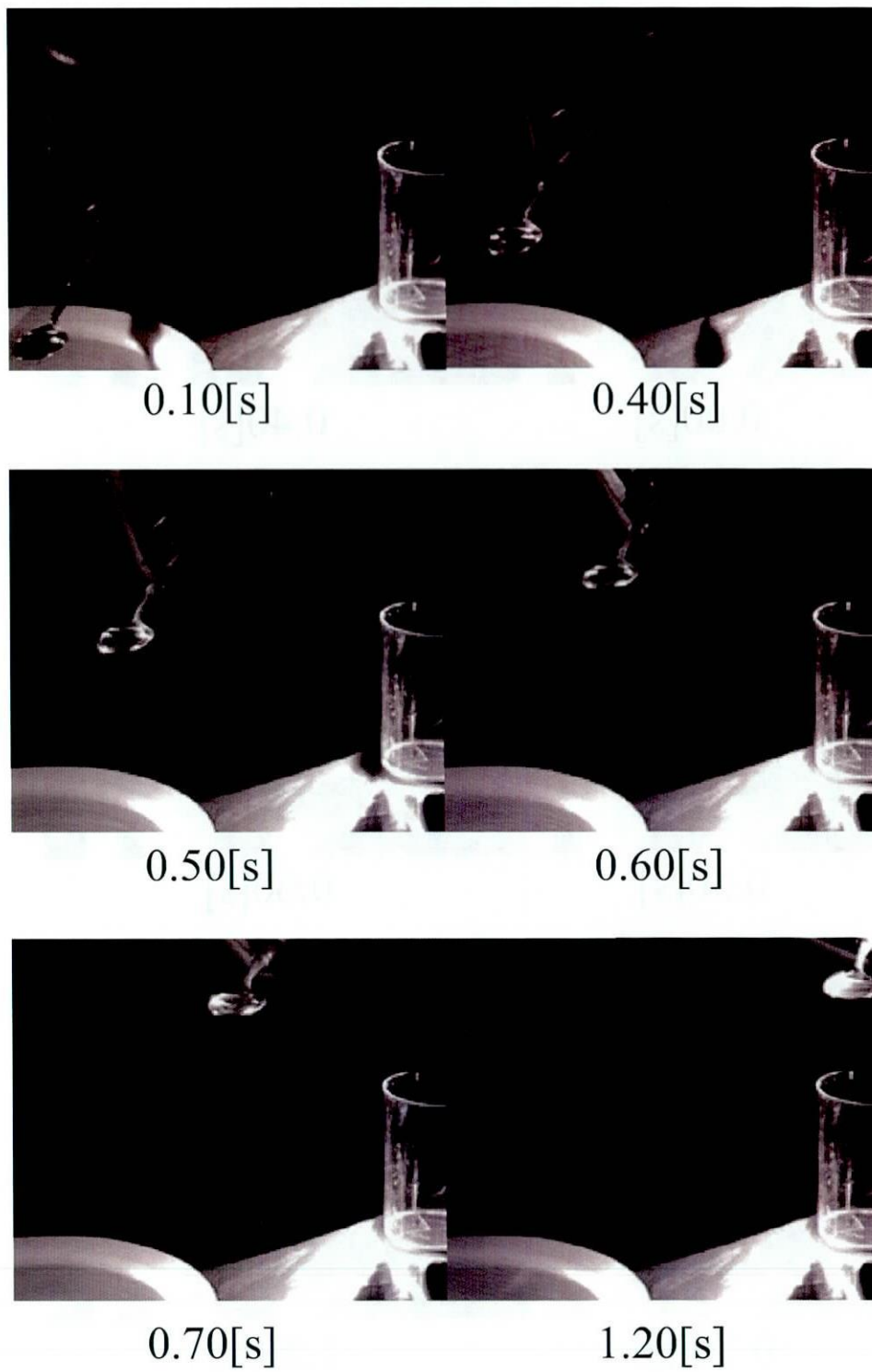


Fig. 2.40 Experimental result by the proposed method

2.7 まとめ

本研究ではCFDシミュレータを援用した最適化として、掬った液状食物を溢さずに搬送することを可能とする液体ハンドリング制御システムの開発を行った。最適化問題では溢流をペナルティ項として定式化し、すばやく、かつ液体の溢流を抑制する搬送軌道の導出を行った。このとき最適化アルゴリズムには、もっとも広く使われている遺伝的アルゴリズムを適用して最適化を行った。結果として、制御系設計に流体解析シミュレータを援用した最適化問題を組み込むことにより、振動制御系や位置決め制御系では困難であった小型容器による溢流抑制液体搬送を実現した。

第3章 ダイカストプランジャの最適射出速度入力の導出

3.1 概要

現在、自動車をはじめとする様々な製品の製造現場で、ダイカストが行われている。ダイカストとはプランジャで型に溶湯を押し込んで製品を生産する鑄造方法のことを指す。ダイカストは大量生産に適した鑄造法であり、寸法精度よく複雑形状を成型することができ、さらに鑄肌がきれいで、機械加工を必要としないため、砂型鑄物と比較して材料を節約することができるなどの多くの利点を持つ。しかしながら、プランジャによる圧入で金型に溶湯を高速充填するため、空気の巻き込みによる鑄巣（巻き込み巣）が発生し、製品の強度にばらつきが出るなどの欠点がある。

このようなダイカストの欠陥防止対策として、プランジャの速度制御があげられる。ダイカストマシンの進歩により、多様なプランジャ速度制御が可能になっているが、製造現場では従来までの、前半部分を低速、後半部分を高速に設定する2段階速度制御に留まっているのが現状である。これは鑄込み動作及び、凝固現象が金型内部の可視化できない場所で発生することから、プランジャ速度設定を技術者の経験と勘のみに頼っており、より多くのパラメータ調整が必要となる多段階速度設定を手動で行うことが困難なためである。そのため、巻き込み巣の発生を防止する最適な圧入が行われているとは言えず、また、このような試行作業により生産準備期間やコストの増加が大きな問題となっているため、より合理的な制御方法の開発が必要とされている。現在ではダイカストにおいても高品質化が望まれており、より適切なプランジャ速度制御を行うことは必要不可欠である。

従来までにダイカスト鑄造における製品欠陥に関する研究は多く行われ、水モデルを用いて射出条件が与える製品品質への影響の調査やダイカストスリーブ内の冷却と凝固現象に伴う流動課程の調査を行うなど様々な研究^{[87]~[92]}がなされた。一方で近年では、鑄造品の品質と生産効率の向上を目的として、鑄造CAEを活用した研究も多く行われてきている。鑄造CAEを使用することにより凝固や湯流れ、熱伝導などの複雑な解析が可能となることを利用し、鑄造CAEの妥当性を評価や最適製造条件を自動的に導出する研究などが行われた^{[45]~[48]}。しかしながら、スリーブ内の溶湯の流れが引き起こす空気巻き込みによる欠陥形成などは解明されていない部分が多く、さらなる解析を行う必要がある。また、巻き込み巣の発生を防止することのできるプランジャ速度の最適制御に関する問題は未解決であり、早急な解決が望まれている。

そこで本研究では、鑄造CAEを援用し、スリーブ内におけるプランジャの移動が引き起こす湯流れ挙動及び、空気巻き込み現象を解析し、その製品品質への影響を評価

する。そして、スリーブ内での空気巻き込み量を低減し、ダイカスト中の巻き込み巣の発生を防止することのできるプランジャ速度の最適化を行う。本研究が実現すれば射出条件の設定などの期間短縮による人件費の削減や、品質のばらつきを抑えることで、無駄のない生産をすることが期待できる。

3.2 ダイカストにおける射出速度の最適化

3.2.1 実験装置

本研究では、Fig. 3.1に示す、宇部テクノエンジ製コールドチャンパダイカストマシン(NX500C)を使用する。このダイカストマシンは、速度制御が可能であり、実際の製造工程においては、前半部分を低速、後半部分を高速に設定する2段階速度切替方式を用いている。この切替方式は、欠陥を減少させる方法として一般的であり、多くの製造メーカーでもこの方式を採用している。本装置では、射出速度に関して、 $0.02 \sim 5.0[\text{m/s}]$ まで設定可能である。また、加速度に関しては、 $-4.23 \sim 4.61[\text{m/s}^2]$ まで設定可能である。金型には最薄部が $0.005[\text{m}]$ のテストパーツを用い、スリーブには内径 $0.075[\text{m}]$ 、全長は $0.44[\text{m}]$ のものをを用いた。ここで、プランジャチップ形状は平型とする。金型、スリーブ及びプランジャチップ材質は熱間工具鋼(SKD61)を、溶湯としてはアルミニウム合金(ADC12)を用いた。鑄込み時の金型温度は $110 \sim 150[^\circ\text{C}]$ (定常状態)、溶湯温度溶解炉内は $660 \sim 680[^\circ\text{C}]$ (スリーブ内溶湯温度は $600 \sim 620[^\circ\text{C}]$)に設定した。なお、離形剤はユシロ AZ7150Wを用いた。

プランジャ速度を低速側 $v_l = 0.5[\text{m/s}]$ とし、高速側を $v_h = 1.0[\text{m/s}]$ 及び、 $2.0[\text{m/s}]$ の2条件で2段階速度切替方式を用いて射出実験を行った。高速切替位置は溶湯がゲート部に到達する位置である $0.35[\text{m}]$ に設定した。Fig. 3.2にブリスタ試験後の試験片の鑄肌写真を示す。ここで、ブリスタ試験とは、試験片を炉で熱することにより、内部の空気を膨張させ、鑄巣の調査をする試験のことである。

結果として、ブリスタ試験後の試験片には、両条件とも多くの鑄巣を含んでいることが確認できる。この欠陥の原因としては、スリーブ内でのプランジャの移動過程及びキャビティ内充填過程における空気巻き込みや離型剤との反応ガス生成、さらには溶湯の凝固収縮時におけるポロシティなどが考えられる。本研究では、特にスリーブ内でのプランジャの移動過程における空気巻き込み現象に焦点をあて、スリーブ内の溶湯の流れが引き起こす空気巻き込みによる欠陥形成の原因を解析し、巻き込み巣の発生を低減するプランジャの速度制御を行う。

シミュレーションにおける溶湯においても実験で使用したADC12を、鑄型材料はSKD61を想定し、粘性、凝固、熱伝導を考慮して解析を行った。用いた各種物性値をTable 3.1, Table 3.2に示す。融解潜熱の定義にはエンタルピー法を使用している。

高速射出を伴うダイカストにおいて、内側からの溶湯の補給が十分あるという理由から凝固に伴う収縮を考慮していない。

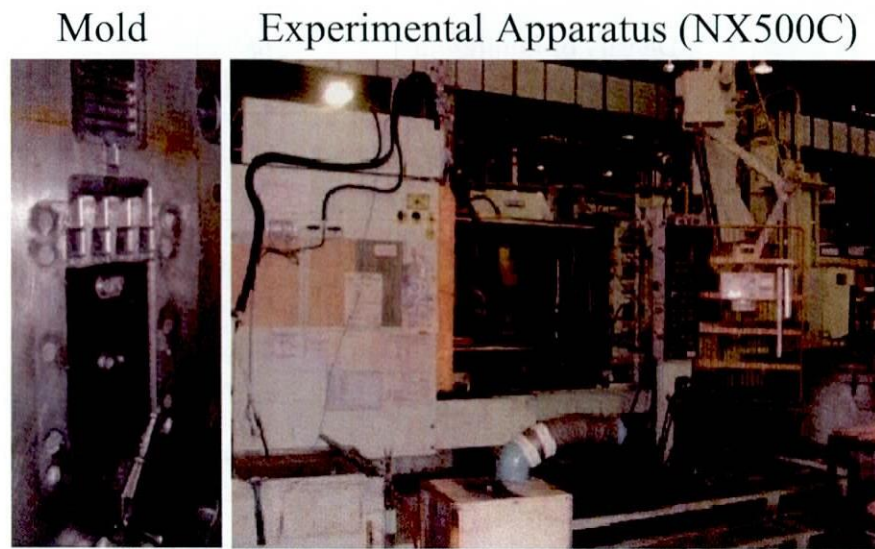


Fig. 3.1 Picture of the cold chamber die casting machine and the mold



Fig. 3.2 Results of a brister test ($v_\ell = 0.5[\text{m/s}]$, $v_h = 1.0, 2.0[\text{m/s}]$)

Table 3.1 Physicality value (Fluid properties ADC12)

Density of fluid[kg/m ³]	2680
Viscosity of fluid[Pa·s]	0.003
Specific heat [J/(kg·K)]	1100
Thermal conductivity [W/(m·K)]	100.5
Initial temperature [K]	953.15
Surface tension [N/m]	0.0723
Latent heat of fusion [J/kg]	397674
Liquidus temperature [K]	861.15
Solidus temperature [K]	803.15
Gravity [m/s ²]	9.8

Table 3.2 Physicality value (Mold properties SKD61)

Density[kg/m ³]	8080
Specific heat [J/(kg·K)]	460.5
Initial temperature [K]	423.15
Heat transfer coefficient [W/(m ² ·K)]	18837

3.2.2 欠陥の解析

空気連行モデル

本研究ではCFDシミュレーションソフトを用いて、スリーブ内におけるプランジャの運動が引き起こす溶湯への空気巻き込み量を推定(以下:空気連行量と表記)する。本解析では、液体表面での空気巻き込みは乱流渦が自由表面のわずかな液体要素を持ち上げるという考えに基づき算出される。つまり、乱れの強さが重力と表面張力の安定勢力を超え、液体要素が自由表面上で持ち上げられる範囲を解析する。ここで、乱流輸送モデルは比乱流運動エネルギー Q と散逸関数 D で特徴付けられ、そのとき乱流渦は Eq.(3.1) で与えられる^{[93][94]}。

$$L = 0.1 \frac{\sqrt{Q^3}}{D} \quad (3.1)$$

高さ L まで持ち上げられた流体要素と表面張力エネルギーに関連した単位体積ごとの外乱運動エネルギー P_d は Eq.(3.2) となる。

$$P_d = \rho g L + \sigma / L \quad (3.2)$$

ここで、 ρ は液体密度、 σ は表面張力、 g は自由表面への重力である。なお、外乱運動エネルギー P_d は、表面張力よりもポテンシャルエネルギーに対し支配的であるため、表面張力による影響は微少と考えられる。

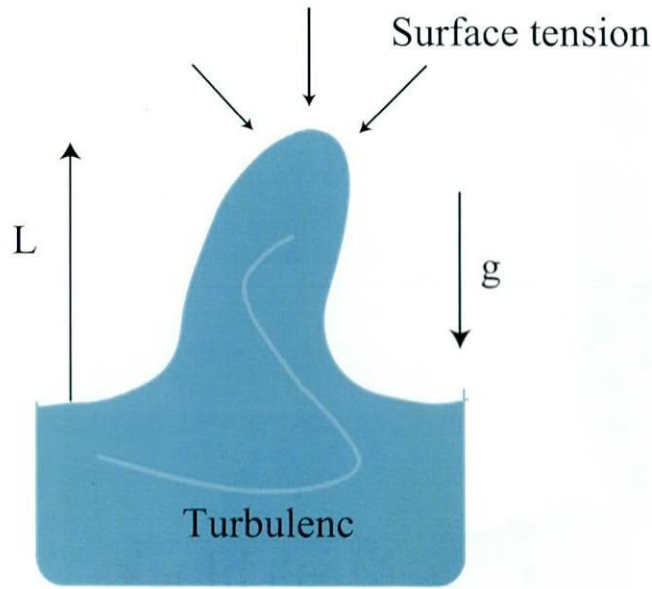


Fig. 3.3 Entrainment model

単位体積あたりの乱流運動エネルギー ($P_t = \rho Q$) が外乱運動エネルギー P_d より大きくなるとき、空気巻き込みが発生し、単位時間に巻き込まれた液体中の空気連行体積率 V_a は Eq(3.3) のように表現できる。

$$\frac{\partial V_a}{\partial t} + u \nabla V_a = R(1 - V_a) \quad (3.3)$$

ここで、 $R = C_{\text{air}} \sqrt{2(P_t - P_d)/\rho}$ であり u は流体速度、 t は時間である。また C_{air} は、空気が捕らえられる面積率を示す係数であり、ここでは、 $C_{\text{air}} = 0.5$ と設定した。なお P_t が P_d 以下であるなら、 $V_a = 0$ となる^[95]。

Eq(3.3) から算出される各メッシュセルでの V_a : 空気連行体積率、 F_f : 流体体積率、 V_f : セル体積率（障害物領域と流体領域との割合）と、各メッシュセルの体積を掛け合わせ合算し、Eq.(3.4) を用いて空気連行量 $a(t)$ を計算する。

$$a(t) = \sum_{k=1}^n V_{ak} F_{fk} V_{fk} V_{ck} \quad (3.4)$$

ここで、 V_c はメッシュセルの体積、 n はメッシュセル総数である。実験において実際にプランジャの移動が引き起こす空気巻き込み量を厳密に計測することは不可能であるため、この空気連行量 $a(t)$ を評価することは困難である。そのため本研究では、解析の結果得られるスリーブ内充填完了時 ($t = t_{\text{fill}}$) の空気連行量 $a(t_{\text{fill}})$ を空気巻き込み易さを表す指標 A として用いる。

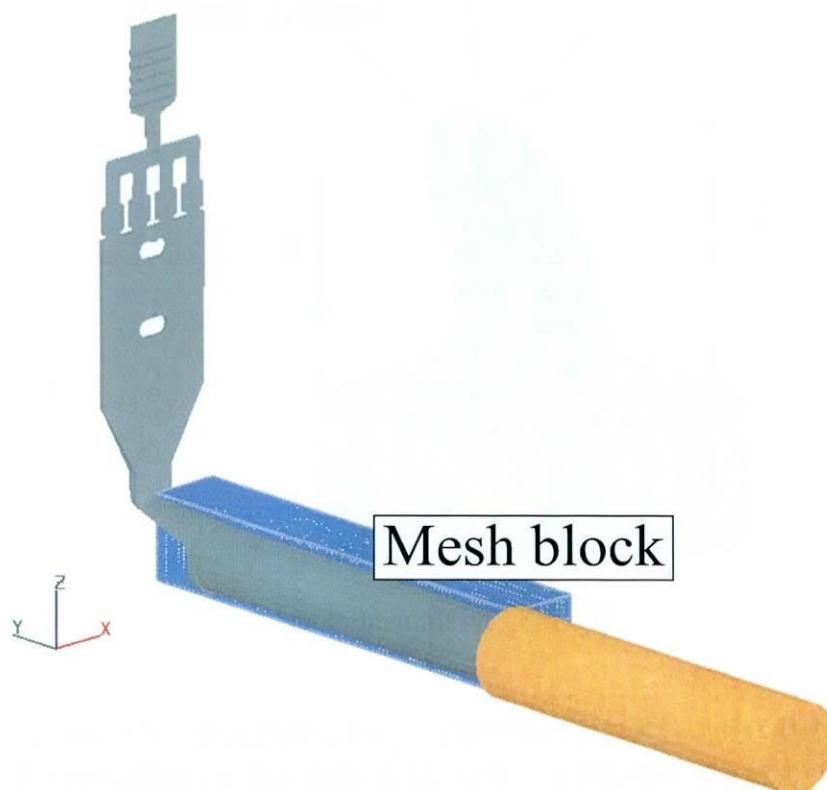


Fig. 3.4 Mesh setting for CFD simulation

CFD シミュレーションによる空気連行量解析

本節ではCFD シミュレーションを用いスリーブ内の空気連行量の解析を行い、低速側速度の変化に対する空気連行量の影響を調査する。シミュレーション領域におけるメッシュ設定図を Fig. 3.4 に示す。スリーブ形状はX軸方向について対称であることから、片側のみを領域設定した。メッシュは流動が緩やかなスリーブ下方(-Z方向)は荒く、液面の流動を伴うスリーブ上方(+Z方向)は細かく、また、プランジャ動作直後の比較的流動が緩やかなスリーブ後方(-Y方向)は荒く、スプルーコア衝突後の波の飛び散りや振り返り波が発生しやすいスリーブ前方(+Y方向)は細かく設定した。スリーブ断面方向(X方向)については一様にメッシュサイズを設定した。なお、金型部分は考慮しない。メッシュパラメータを Table 3.3 に示す。

加速距離を0.05[m]に固定し、低速側速度 v_ℓ を0.20～0.60[m/s]まで0.01[m/s]ごとに变化させてスリーブ充填までの空気連行量の解析を行った。速度毎の空気連行量の解析結果を Fig. 3.5 に示す。1条件当りの計算時間はIntel Core2 Quad 2.83GHz 搭載パソコンで、約5分であった。

シミュレーション結果より、0.37[m/s]以上では Fig. 3.6($v_\ell = 0.50$ [m/s]) に示すように、初期の波の崩れ、スプルーコアへの衝突による飛散、溶湯による空気の閉じ込めが発生していることが確認できる。CFD シミュレーションによる空気連行量の総計 A

Table 3.3 Mesh Parameters for die-casting

	Cell size [m]	Number of cell
X-direction	0.004	20
Y-direction	0.002 ~ 0.006	132
Z-direction	0.0022 ~ 0.0035	29
Total number of cell		76,560

は6.588であったが、 $t = 0.66[\text{s}]$ において閉じ込められた空気は射出時に溶湯中に押し込められるため、実際の連行量は計算結果よりも多くなると考えられる。このことがFig. 3.2に示した欠陥の原因となっていると考えられる。

また一方で、 $0.23[\text{m/s}]$ 以下に減速したとしても、Fig. 3.7($v_\ell = 0.21[\text{m/s}]$)に示すように、振り返りとプランジャとの再衝突により、大きな波が発生するとともに、溶湯による空気の閉じ込めも発生していることが確認できる。CFDシミュレーションによる空気連行量の総計 A は1.598に低減されているが、空気閉じ込め現象が発生しているため、実際の巻き込み量は計算結果よりも多くなると考えられる。このことから、単に低速で射出しただけでは、空気巻き込みの発生を防止し、製品欠陥を抑制することができないことがわかる。

確認のため、 $v_\ell = 0.21[\text{m/s}]$ とし、制御実験を行った。その結果をFig. 3.8に示す。なお、高速側速度は $v_h = 2.0[\text{m/s}]$ と設定した。Fig. 3.8より、 $v_\ell = 0.21[\text{m/s}]$ とした条件において、Fig. 3.2に示した結果よりは、気泡の数は大幅に減少しているが、左右に大きな気泡を、また中央部に細かい気泡が多く発生しており、単純に速度を低下させるだけでは空気巻き込みを防止できないことが確認できる。なお、同様の実験を3回行ったがほぼ同様の結果が得られた。

それに対し、 $0.24 \sim 0.36[\text{m/s}]$ では、Fig. 3.9($v_\ell = 0.26[\text{m/s}]$)に示すように、波立ちが少ない良好な湯流れを示した。CFDシミュレーションによる空気連行量の総計 A は2.547であり、空気閉じ込め現象を伴わないため $v_\ell = 0.50$ 及び $0.21[\text{m/s}]$ と比較して良好な射出が行われているといえる。結果として、スリーブ内で空気を巻き込まないためには、(1)プランジャの初動で波を崩さない、(2)スプルーコアに衝突後、波が湯道を塞がない(波の高さとスプルーコアの高さがほぼ同じ)、(3)空気の閉じ込めを発生させないことが重要であることがわかる。またTable 3.4に示す空気連行量の計算結果から、空気連行を低減し、空気閉じ込めを発生させない良好な射出が行われていることが伺える。

同様に確認のため、 $v_\ell = 0.26[\text{m/s}]$ の条件で射出実験を行った。その結果をFig. 3.10に、また実験におけるプランジャ速度をFig. 3.11に示す。

実験結果より、 $v_\ell = 0.26[\text{m/s}]$ とした条件において、空気連行量の推定値が大きい上記2例の結果と比較し、製品部中央付近ではまだ細かい気泡が見られるものの、表面

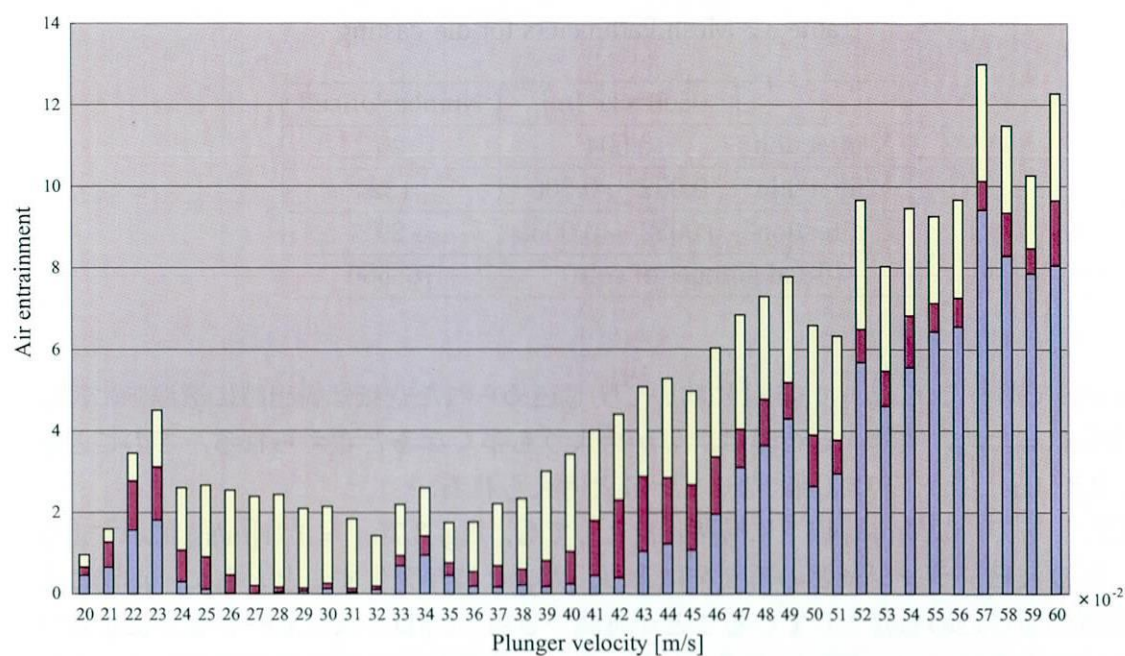
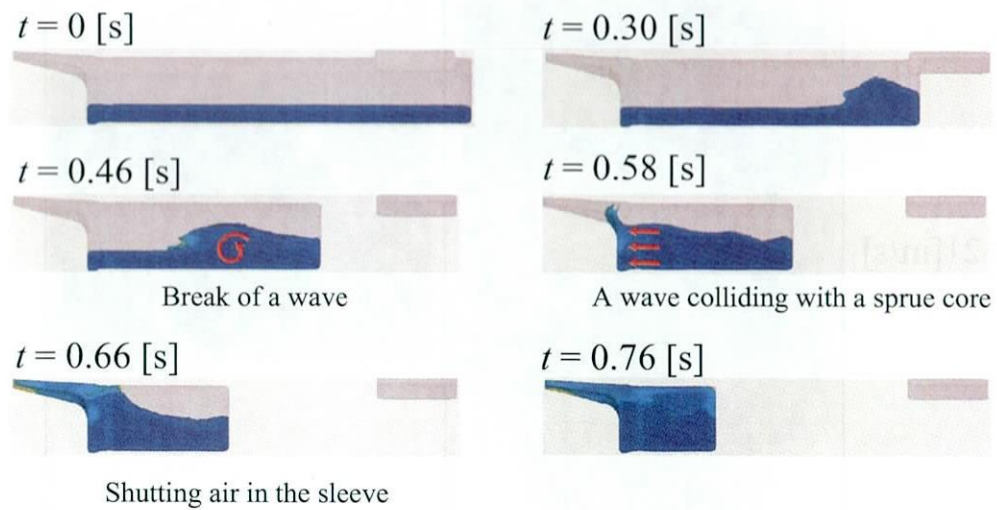
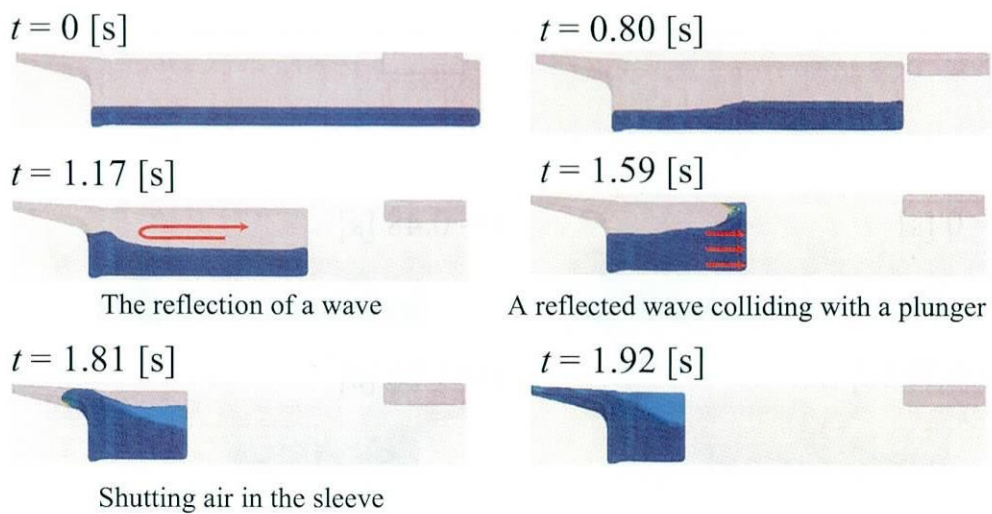


Fig. 3.5 Result of air entrainment searched all velocity

Table 3.4 Quantity of air entrainment

v_l [m/s]	Quantity of air	Air shutting
0.21	1.598	exist
0.26	2.547	no
0.50	6.588	exist

の気泡が明らかに減少していることがわかる。またこのことから本解析手法の有効性が確認できる。本シミュレータを使用し、空気連行量の低減化を可能とするプランジャ速度入力設計ができれば、実際の射出実験においても良好な結果が得られることが予想される。次節以降では、空気連行量を最小化する2段階速度切替方式におけるプランジャ射出速度の最適化を試みる。

Fig. 3.6 Simulation result of $v_\ell = 0.50$ [m/s]Fig. 3.7 Simulation result of $v_\ell = 0.21$ [m/s]

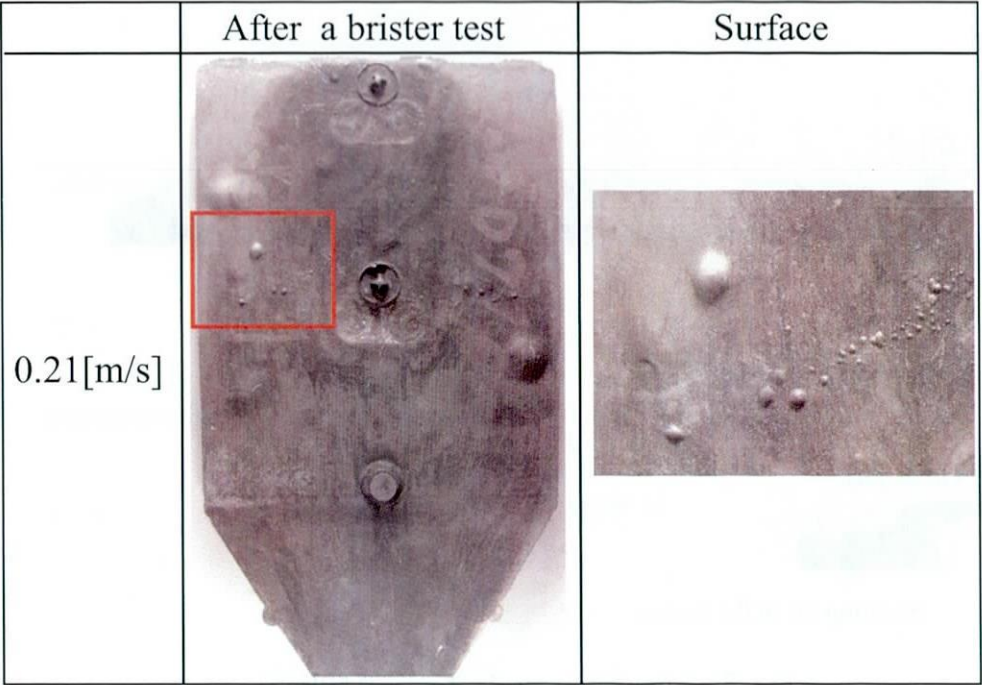


Fig. 3.8 Result of a brister test ($v_\ell = 0.21[\text{m}]/\text{s}$, $v_h = 2.0[\text{m}/\text{s}]$)

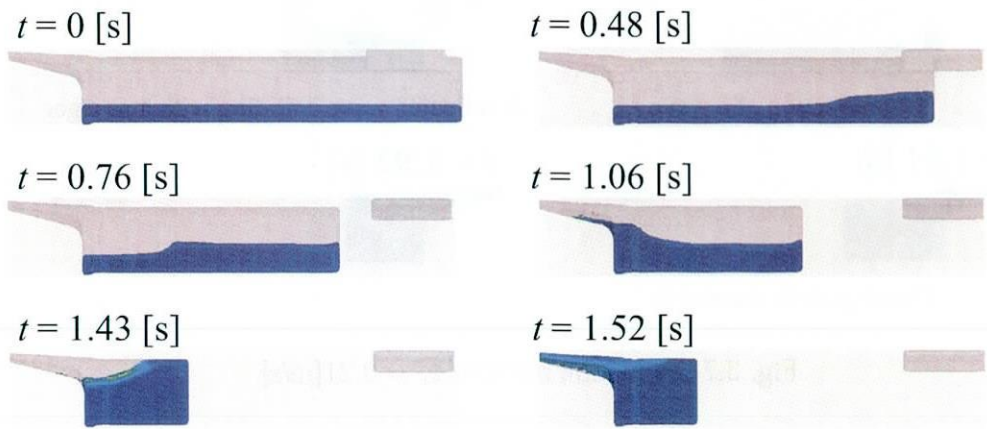


Fig. 3.9 Simulation result of $v_\ell = 0.26[\text{m}/\text{s}]$

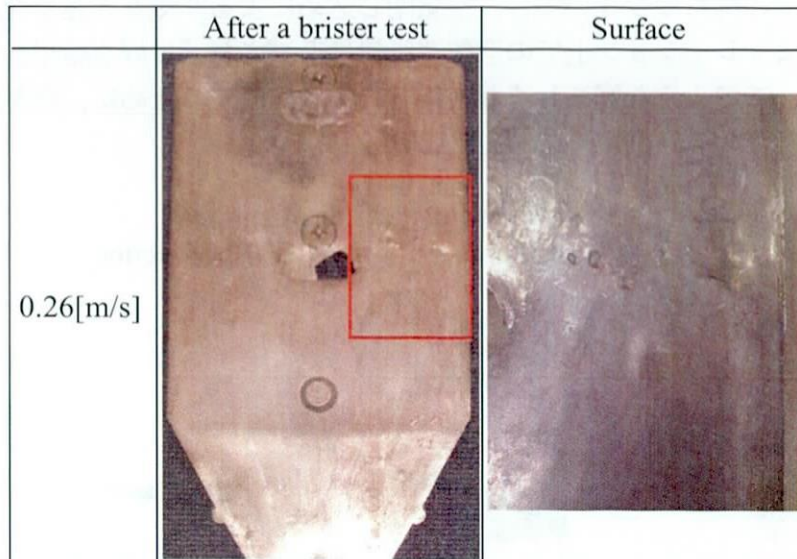
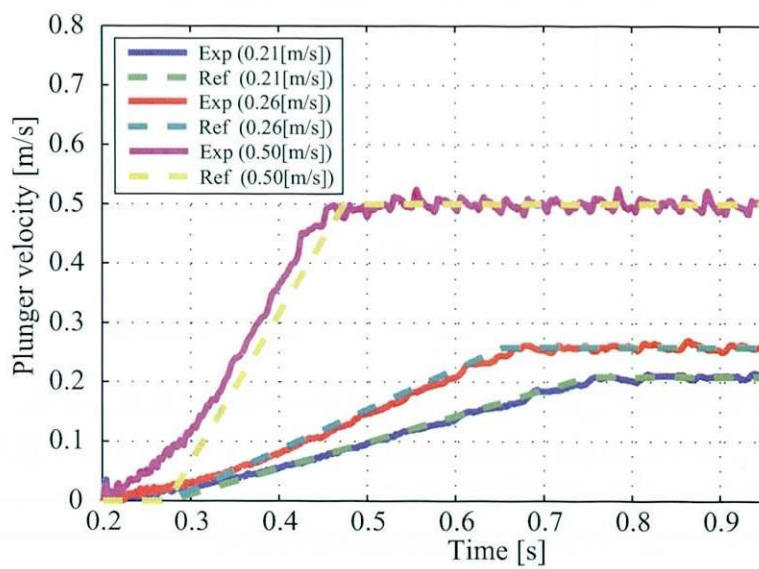
Fig. 3.10 Result of a brister test ($v_\ell = 0.26[\text{m}]/\text{s}$, $v_h = 2.0[\text{m}/\text{s}]$)

Fig. 3.11 Plunger velocities of each result

空気閉じ込め量の計算

プランジャ速度入力によって、Fig. 3.6 に示すような空気閉じ込め現象を伴う溶湯挙動を示すことがある。射出条件によっては空気閉じ込め現象を含む溶湯挙動を示していても、自由表面の乱れが少ない場合、空気閉じ込めによる空気巻き込量を評価していないため、シミュレーション上での空気連行量は低減され『良好な結果』と判断される場合がある。このような好ましくない最適条件を回避するために、空気閉じ込め判別方法を提案した。その概念図を Fig. 3.12 に示す。

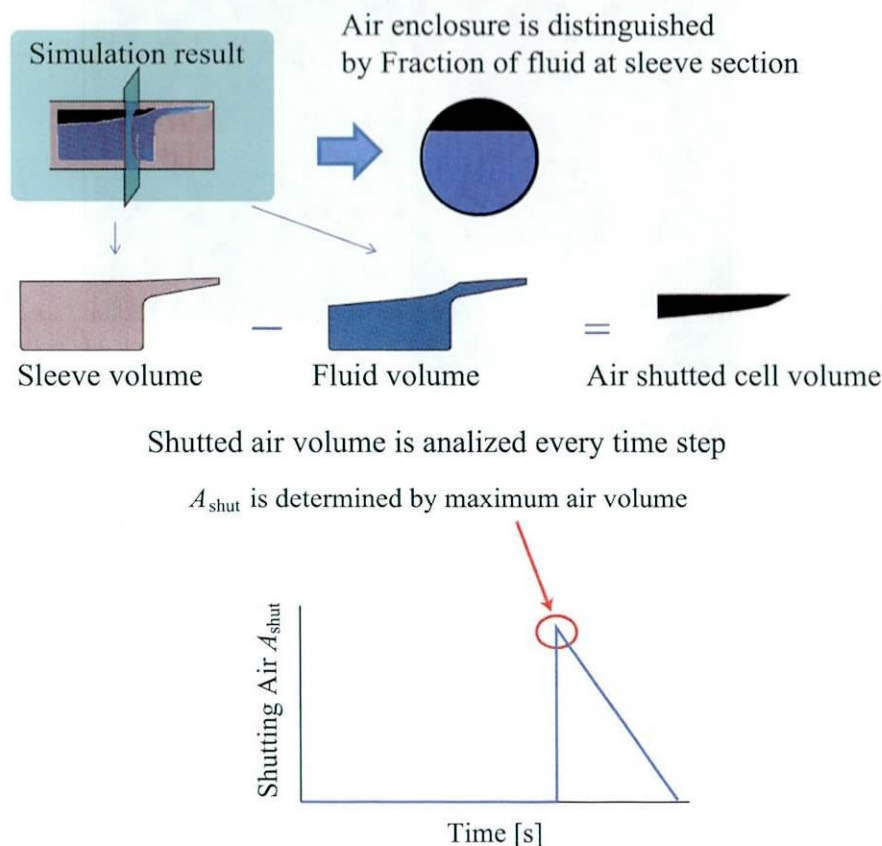


Fig. 3.12 Distinction of the shutting the air in the sleeve

空気閉じ込め量計算のために、以下の3つのパラメータを新たに導入する。Flow-3Dのカスタマイズは、これらを prepin の Namelist;xput に入力することにより適用される。

- D_1 : Y断面における流体の体積/開口体積
- D_2 : メッセージファイルへ表示する空気巻き込み量の閾値
- D_{25} : 出力ファイルの時間間隔

D_{25} で指定した時間間隔ごとに CFD シミュレーションを用いてセルの体積率 (Volume Fraction), 流体率 (Fraction of fluid) を出力し、スリーブ断面毎に充填率を計算する。スリーブ断面に対する流体率が $D_1 \times 100[\%]$ の部分よりも後方側の空間体積を計算し、最大の空間体積を空気閉じ込め量 A_{shut} として定義する。カスタマイズを使用すること

で、内部計算による空気閉じ込め量算出ができるため、3次元で空気閉じ込め体積を出力しても、計算時間は標準の解析と比較してほとんど変わらない。以降の解析には全てこのカスタマイズを用い、空気閉じ込め量 A_{shut} を評価値及び制約条件として使用する。

3.2.3 評価関数の構築

ダイカストにおけるプランジャ速度の設定は Fig. 3.13 に示す5変数の速度設定を用い最適条件を導出する。ここで、プランジャ速度 v_1, v_2, v_3 , 加速距離 x_1, x_2 の5変数を可変パラメータとし、 x_{fill} ：充填完了位置は一定である。

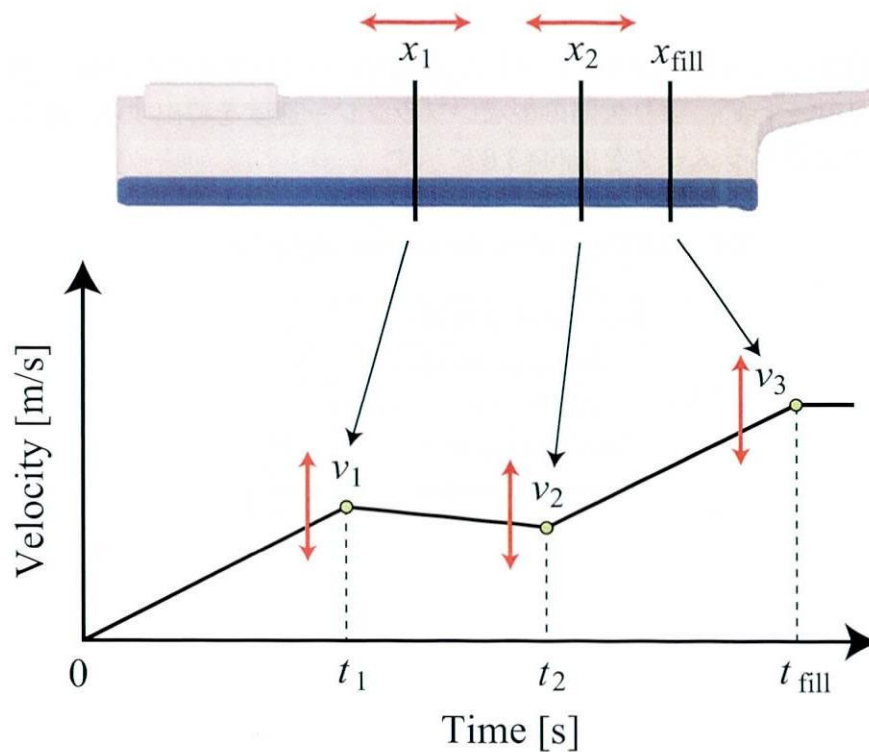


Fig. 3.13 Velocity curve of a plunger

前節までのCFDシミュレーション結果及び実験結果より、空気巻き込みが欠陥発生の大きな原因となっていることがわかった。また一般的に充填に時間を要すると溶湯温度、固相率に変化が現れる。そこで、空気連行量及び、充填完了時間をそれぞれ重付けし、それらの和を評価関数とし、Eq.(3.5)に示すように最適化問題を定式化した。

$$\text{minimize : } J = w_a A(v(t), x) + w_t t_{\text{fill}}(v(t), x) + K_p \quad (3.5)$$

$$\begin{aligned} \text{subject to : } & 0.02 \leq v \leq 0.60 \\ & 0.02 \leq x \leq 0.36 \\ & 0 \leq t \leq 2.0 \\ & A_{\text{shut}} \leq 2.0 \end{aligned} \quad (3.6)$$

ここで、 A :空気連行量, T :充填完了時間 [s], x :速度切換え位置 [m], w_a :空気連行の重み, w_t :時間重み, A_{shut} :空気閉じ込めである. 最適条件付近での空気連行量, 充填完了時間の影響が同程度になるように, 評価値重みを Table 3.5 のように設定する. ここで重みに関しては, 評価関数中の空気連行量 A が最小となり, 同時に充填完了時間 t_{fill} ができる限り小さくなるように設定した.

Table 3.5 Weight of evaluation value

w_a	1.5
w_t	1.0

次節では, CFD シミュレーションソフトと遺伝的アルゴリズムを用いることにより, 評価関数の最小化を行い, より欠陥の少ないプランジャ速度を導出する. なお, 最適化計算に用いた計算パラメータを Table 3.6 に示す.

Table 3.6 Parameters for Genetic algorithm

Number of variable	2
Number of population	30
Number of elite preservation	1
Mutation fraction	0.01
Crossover fraction	0.80

3.3 シミュレーション結果

最適化計算結果より、各世代毎の評価値のグラフを Fig. 3.14 に示す。計算時間は約 25 時間であった。

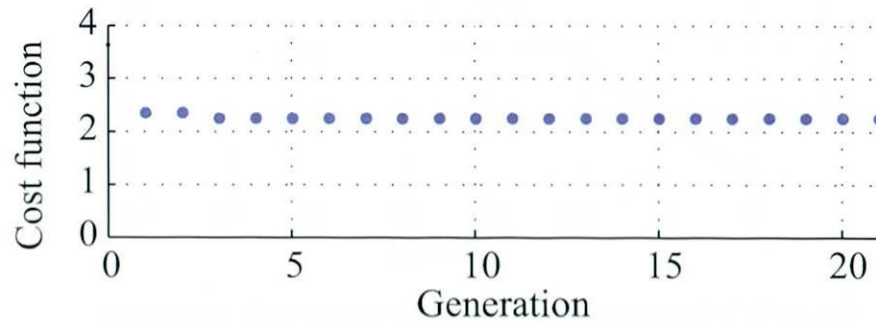


Fig. 3.14 Optimization result

導出された最適入力及びアニメーションを Fig. 3.15, Fig. 3.16 に示す。このとき、最適入力時の評価値は $J = 2.24$, $A = 0.53$, $t_{\text{fill}} = 1.44[\text{s}]$ であった。

導出された射出条件を解析すると以下の動作が行われていることがわかる。

1. プランジャ動作直後の波の崩れが発生しない
2. 1 回目の加速時に波が発生し進行する
3. 一旦減速した後再び加速を始め、新たな波を発生させる
4. 反射した波と進行する波が衝突することにより、プランジャと反射波の衝突を防ぎ、空気巻き込みを低減する。

5 変数を用いることにより、2 回の加速を行うことにより、波を打ち消しあう良好な射出条件を導出した。波を 2 回発生させることによりスリーブ充填完了時間は文献^[96]において導出された最適条件と比較し、約 24 % の短縮が可能となった。次節において、実験を行い導出された最適速度入力の検証を行う。

3.4 実験結果

求まった最適速度入力を用いて実験を行った。比較対象として、同手法で求めた 2 変数最適条件を用いる。用いた 5 変数の速度入力を Table 3.7 に、2 変数の速度入力を Table 3.8 に示し、プリスタ試験結果を Fig. 3.17 及び、Fig. 3.18 に示す。結果として、試験片表面に Fig. 3.2 に見られるような大きな気泡は見られず、5 変数を用いることにより、2 変数よりも短時間で同程度の空気巻き込み低減できていることがわかる。このことから最適化手法の妥当性が証明でき、また変数を増大することで、空気巻き込み量がより低減できることがわかった。

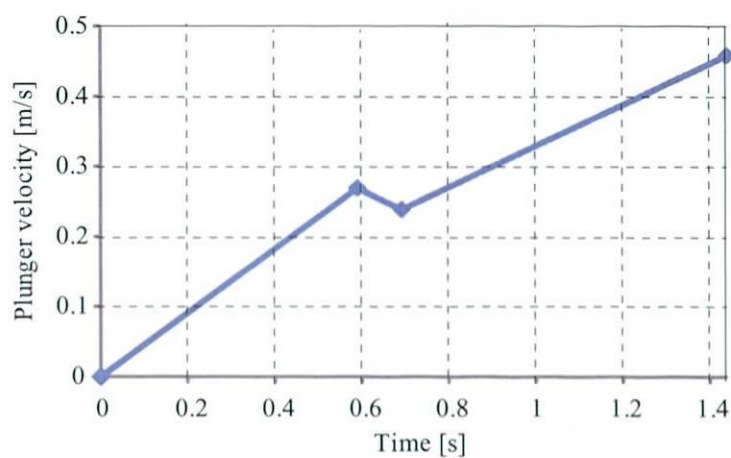


Fig. 3.15 Velocity curve of optimum velocity with 5 variables

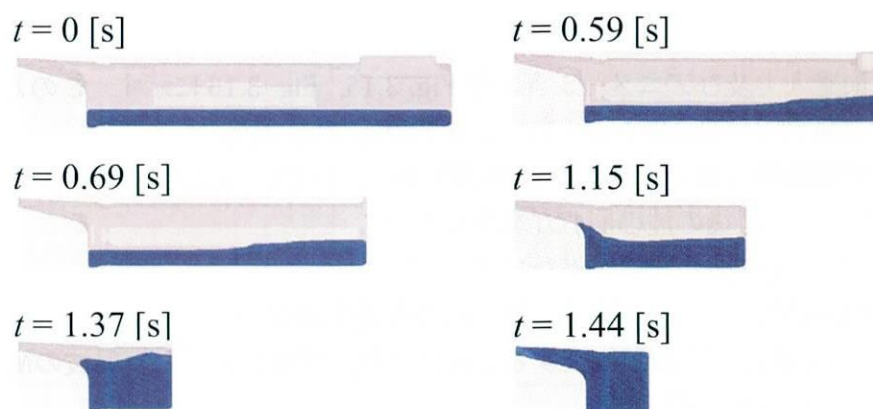


Fig. 3.16 CFD simulation result using optimum velocity with 5 variables

Table 3.7 Optimum plunger velocity

5 variables	Time [s]	Velocity [m/s]	Position [m]
$i = 1$	0.59	0.27	0.079
$i = 2$	0.69	0.24	0.105
$i = 3$	1.44	0.46	0.367
Cost function : 2.24			
Total area of the air bubble : $3.63 \times 10^{-4} \text{ [m}^2\text{]}$			

Table 3.8 Optimum plunger velocity

2 variables	Time [s]	Velocity [m/s]	Position [m]
$i = 1, 2$	1.63	0.32	0.260
$i = 3$	1.96	0.32	0.367
Cost function : 2.62			
Total area of the air bubble : 7.52×10^{-4} [m ²]			

3.5 まとめ

本章ではCFDシミュレータを援用した最適化として、ダイカストを対象とした製品品質向上を目的としたプランジャ速度の最適化を行った。シミュレータを用いて欠陥原因の解明を行うとともに、実験結果とシミュレーション結果が一致するような評価関数を構築した。また最適化アルゴリズムには解空間が多峰性であることから、もっとも広く使われている遺伝的アルゴリズムを適用して最適化を行った。結果、最適化を行うことでシミュレーションおよび実験結果ともに良好な結果を得ることができ、空気巻き込み量を低減することができた。

またCFDシミュレータを援用した最適化の応用例として著者は、遺伝的アルゴリズムを用いて重力金型鑄造における最適注湯速度の最適化を行った。重力金型鑄造における最適化では、実験では金型を対象とするもののシミュレーションでは型の流動は考慮せず、湯だまり内の液位をすばやく高く一定に保つ入力を求めることで、品質の向上を行った。この詳細は、CFDシミュレータを援用した最適化の応用事例として付録Cに示しておく。

CFD最適化問題には解探索に膨大な計算時間が必要であったため、必然的にその探索数を少なくした。しかしながら遺伝的アルゴリズムは多くの個体数(探索点)が使えるという前提で設計されているため、少ない個体数で解を見つける場合は非常に局所解に陥りやすくなる。特に、ダイカストのプランジャ速度の最適化においては、評価関数に空気連行量のようなCFDシミュレータから得られた数値を取り入れる場合、桁落ちや情報落ちの影響により解空間はさらに多峰性を帯びる。そのため、さらに局所解に陥りやすい空間となり、数値的に良好であっても実験結果では良好な結果を得難いことが問題となった。したがってCFDシミュレータを援用した最適化を実用化するには、良好な実験結果を得られるような解を導出することのできる最適化アルゴリズムが必要である。



	After a brister test	Surface
$v_1 =$ $0.27[m/s]$ $v_2 =$ $0.24[m/s]$ $v_3 =$ $0.46[m/s]$ $x_1 =$ $0.079[m]$ $x_2 =$ $0.105[m]$		

Fig. 3.17 Result of a brister test using optimum velocity with 5 variables

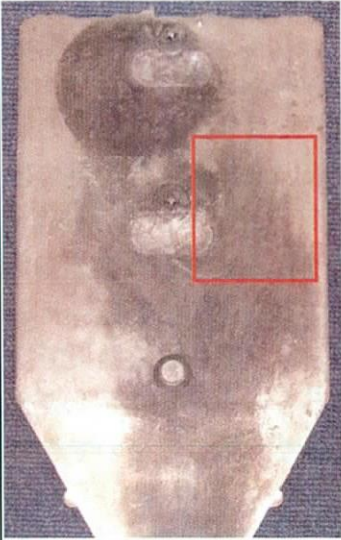

	After a brister test	Surface
$v =$ $0.32[m/s]$ $x =$ $0.26[m]$		

Fig. 3.18 Result of a brister test using optimum velocity with 2 variables

第4章 極値近傍探索アルゴリズム

4.1 概要

良好な実験結果を得られるような解を導出するには、これまでの最適化のように、最良の評価をもつ解ではなく、シミュレーションと実験とのモデル誤差の影響を受けにくい、大域的に良好な評価を得ている解を見つける必要がある。つまり解周辺の評価も考慮に入れ、大域的に良好となっている解空間の中心を見つけることができれば実験においても良好な結果を見つけることができる。また良好な解空間や劣悪な解空間など、解空間を把握することができれば、良好な解空間を判断し集中的に個体を配置させることで、探索性能の向上が期待できる。そこで本研究では、近似曲線と極値を利用した解探索法である極値近傍探索アルゴリズム (Extremal Distribution Sorting Algorithm : 以下 EDSA と表記) を提案する。具体的には近似曲線と極値を利用することで、解空間の予測をすることができ、さらに良好な空間への集中探索や大域的安定解の算出をすることが可能となる。Fig. 4.1 に本研究で提案する EDSA の概要を示す。本手法の有効性を、ダイカストにおけるプランジャ速度の最適化に適用して、実験結果を通して示すことにする。

4.2 極値近傍探索アルゴリズムの構築

4.2.1 近似曲線の導出

CFD シミュレータで得られた解を変数ごとにプロットし、その近似曲線と極値から解の傾向を探索する。解の探索に用いる近似曲線は最小二乗法により導出する。Eq. (4.1) に近似曲線を導出する式を示す。

$$\begin{pmatrix} N & \sum_{i=0}^N x_{i,k} & \cdots & \sum_{i=0}^N x_{i,k}^m \\ \sum_{i=0}^N x_{i,k} & \sum_{i=0}^N x_{i,k}^2 & \cdots & \sum_{i=0}^N x_{i,k}^{m+1} \\ \cdots & \cdots & \cdots & \cdots \\ \sum_{i=0}^N x_{i,k}^m & \sum_{i=0}^N x_{i,k}^{m+1} & \cdots & \sum_{i=0}^N x_{i,k}^{2m} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \cdots \\ a_m \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^N y_i \\ \sum_{i=0}^N x_{i,k} \cdot J_i \\ \cdots \\ \sum_{i=0}^N x_{i,k}^m \cdot J_i \end{pmatrix} \quad (4.1)$$

ここで、 N : サンプル数, n : CFD 最適化問題の次数, m : 近似曲線の次数, $x_{i,k}$: k 変数, i 番目の個体の解, J_i : i 番目の個体の評価値とする。

近似曲線は、 n 次の変数の CFD 最適化問題に対して、評価値と 1 変数による 2 次元空間から導出する。その導出を n 回繰り返すことにより、それぞれの変数に対して、解の傾向を探索する。近似曲線を生成するためのサンプルの数 N は、シミュレーション

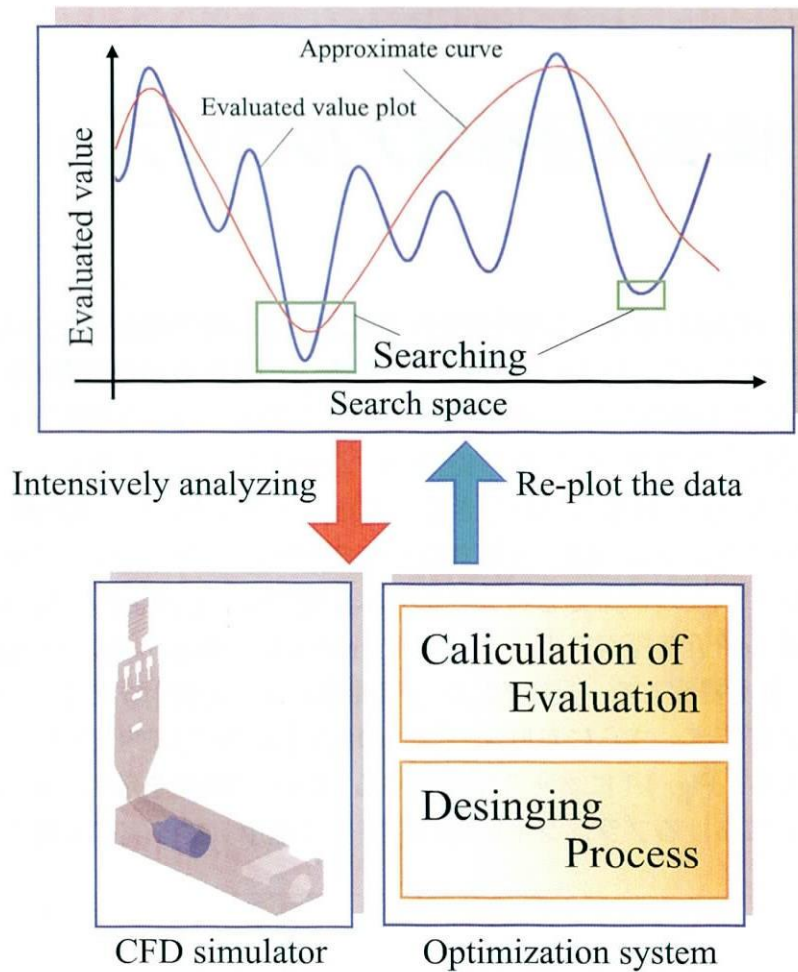


Fig. 4.1 Concept of the EDSA

モデルや解析の進行具合によって増減するため、サンプル数に応じて曲線の次数 m を変化するようにアルゴリズムを構築する。また最適化計算が進むにつれて同じ解構成を持つ個体が増加することが予想されるため、実質的なサンプルの数が減少する。サンプル数の減少も視野に入れ、近似曲線の次数は調整される。なお、このアルゴリズムで採用する近似曲線の最大次数は5次曲線とした。

4.2.2 極値の判定

近似曲線を用いる場合、問題に対して近似曲線の次数が正しく設定されなかった場合、解空間が大きく変わってしまい、空間予測も誤ってしまう。そこで近似曲線による推移だけでなく、基の評価値と近似曲線の極値を比較し、良好な解空間の判定を行う。Fig. 4.2 に5変数最適化問題の評価値から近似曲線を導出した例を示す。

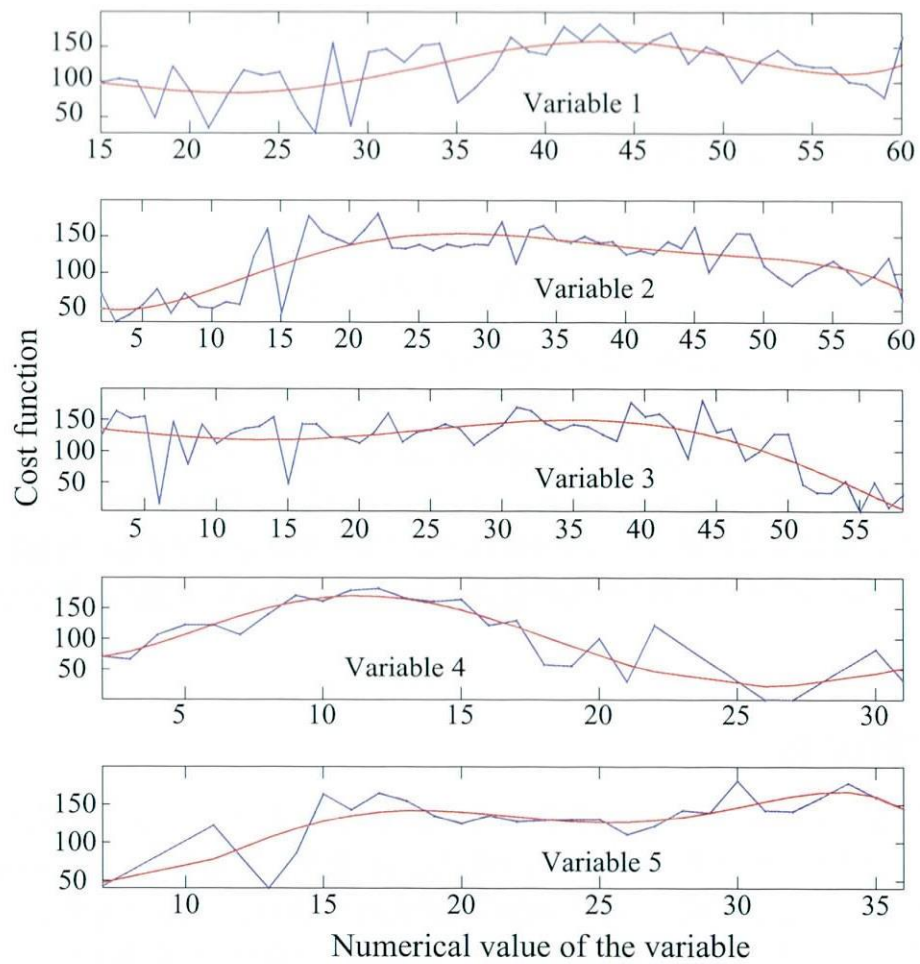


Fig. 4.2 Extremum plot example

Fig. 4.2 はダイカストプロセスの5変数速度設定のCFD最適化問題の解のプロット図である。評価値と1変数を2次元プロットし、それぞれに対して極値と近似曲線を導出する。Fig. 4.2 は評価値が低いものほど良好な解であることを示している。解空間の予測では評価値とそれによって形成された近似曲線、それぞれの別々で極値の近傍を求める。

CFD シミュレータの評価値を $f(x)$ とすると、 $f(x)$ は不連続なので微分による極値の導出はできない。そのため、 k 変数、 i 番目の個体 $x_{i,k}$ が極値であるか否かの判断基準は、

$$\begin{aligned} (f(x_{i+1,k}) - f(x_{i,k})) \times (f(x_{i,k}) - f(x_{i-1,k})) &< 0 \\ 0 &< (f(x_{i,k}) - f(x_{i-1,k})) \end{aligned} \quad (4.2)$$

を同時に満たすとき、 $x_{i,k}$ は極大値、

$$\begin{aligned} (f(x_{i+1,k}) - f(x_{i,k})) \times (f(x_{i,k}) - f(x_{i-1,k})) &< 0 \\ (f(x_{i,k}) - f(x_{i-1,k})) &< 0 \end{aligned} \quad (4.3)$$

を同時に満たすとき、 $x_{i,k}$ は極小値として判断する。

$x_{i,k}$ を極値として判断したとき、 x_i を極値として保存する。近傍は変数を2次元プロットしていることから、 $x_{i,k}$ の極値近傍を考えると、その変数の最小単位 e_k を用いる。 $x_{i,k}$ の隣り合う2点 $x_{i,k} + e_k, x_{i,k} - e_k$ を $x_{i,k}$ へ代入した個体を生成する。

近似曲線も同様にして極値とその近傍を算出し、その個体を保存する。評価値とその近似曲線で求められた2種類の極値の情報を基に、次節において次世代個体の選出を行う。

4.2.3 次世代個体

評価値およびその近似曲線から2種類の極値を導出した上で、それらの傾向から次世代の親個体候補を選出する。具体的に評価値の高い個体と解の向上が見込められる個体のみを選出候補として残し、その中から次世代の親個体を選出する方法を採る。

まず、CFD シミュレータの解析過程で制約条件を満たせなかった個体をペナルティリスト X_p とする。さらに評価値および近似曲線の劣悪な極値とその近傍の集合を X_b と仮定し、これも同様にペナルティリスト X_p として加える。

次に評価値と近似曲線の良好な極値とその近傍を集合 X_g へ保存する。このとき、良好な極値であったとしても評価値の低い極値については候補から除外する。そのため、極大値の平均値 $\bar{f}(x_g)$ より評価値の大きい極値、

$$f(x_g) > \bar{f}(x_g) \quad (4.4)$$

を満たす極値 x_g のみを候補とし、集合 X_c として保存する。この親個体の候補 X_c とペナルティリスト X_p を照会する。ペナルティリストと一致する個体があれば、候補から

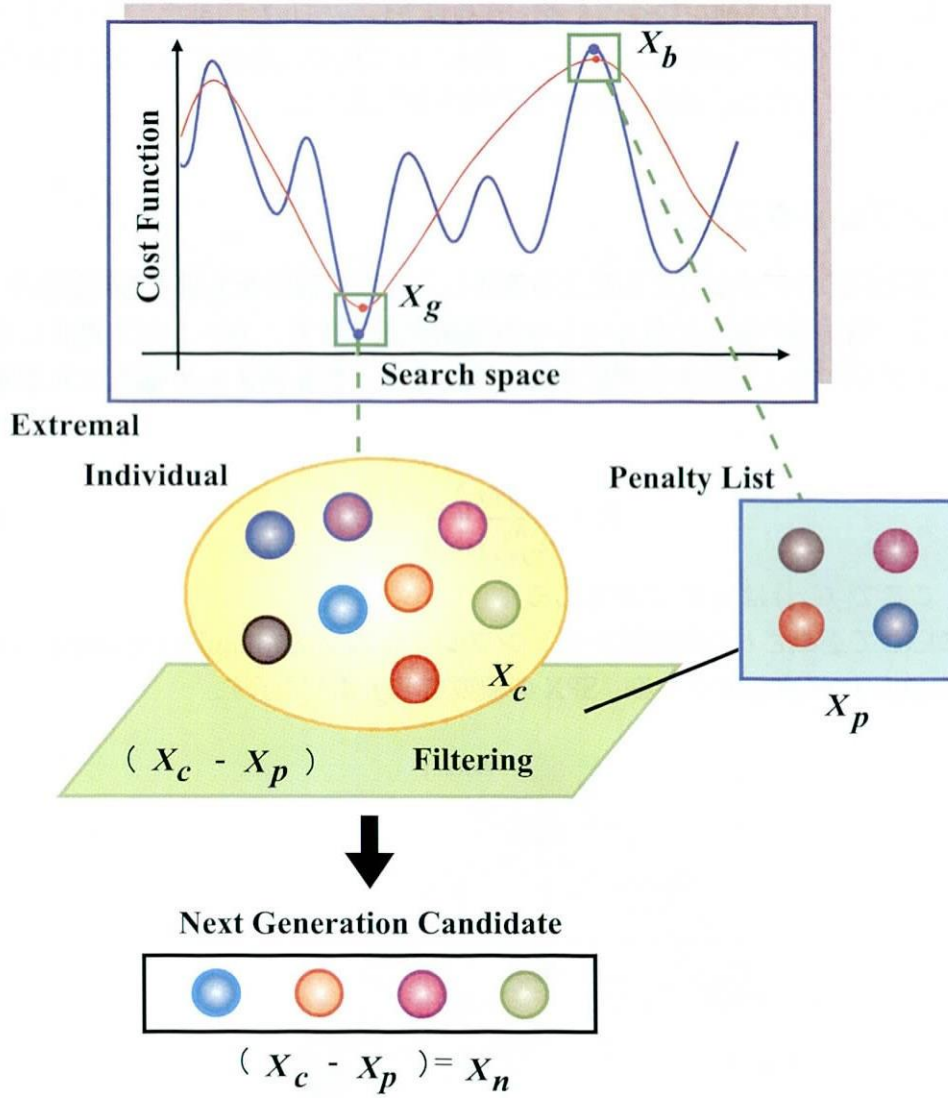


Fig. 4.3 Individual selection

除外していく。候補に残った個体を 4.2.4 小節において次世代個体を集合 X_n に保存する。その概念図を Fig. 4.3 に示す。

評価値の良好な極値を x_g とし、 x_g から Eq.(4.5) で選出された個体を x_c として候補に残す。近似曲線における評価値の劣悪な極値を x_b ，それぞれの近傍を x_{g+e} ， x_{b+e} とする。近似曲線の良好な極値を x_{Ag} ，劣悪な極値を x_{Ab} ，その近傍を x_{Ag+e} ， x_{Ab+e} とする。また評価値の低い個体における極大値については候補から除外するため，Eq.(4.4) を満たす極値 x_c のみを候補に残す。その上で以下の優先順位で候補を選ぶ。

$$\begin{aligned}
 x_c &> x_b > x_{g+e} > x_{Ag} > x_{b+e} > x_{Ab} \\
 &> x_{Ag+e} > x_{Ab+e}
 \end{aligned}
 \tag{4.5}$$

Eq.(4.5) では、極大値の候補になるべく多く残るように設定した。また、解の向上を持続させるため、1世代中の最良解をエリート個体として候補に追加する。これらの個体を親個体候補として保存し、次節において子個体を生成する。

4.2.4 シンプレックス交叉

親集団候補を前節のアルゴリズムにより選出し、次世代個体を生成する親個体を X_n から選出する。親個体の選出にはルーレット選択を適用する。ルーレット選択は評価値に応じた選択率に沿って個体を選択する方法である。ある個体 x_i が選択される確率 P_i は、

$$P_i = \frac{f(x_i)}{\sum_{j=1}^{N_g} f(x_j)} \quad (4.6)$$

となる。ここで $f(x_i)$ は x_i の評価である。

選択率に応じて選出された親個体からシンプレックス交叉 (Simplex crossover：以下 SPX) により次世代個体を生成する。SPX の概要を Fig. 4.4 に示す。

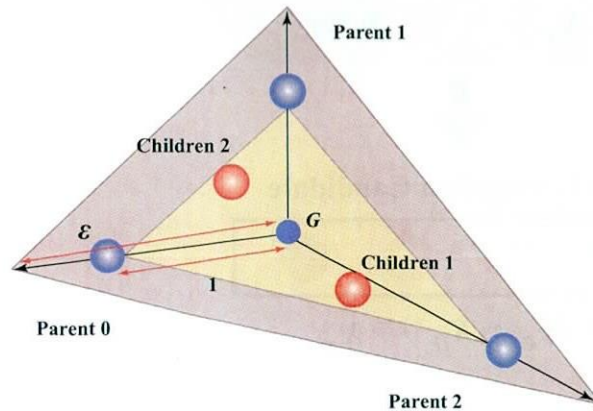


Fig. 4.4 Simplex crossover

SPX とは遺伝的アルゴリズムの一種である実数値 GA の交叉手法である^[97]。実数値 GA は遺伝的アルゴリズムのビットストリングによる交叉手法ではなく、変数を実数ベクトルとして扱う。そのため、特に連続最適化問題に有効な手法であり、変数間の依存関係に強く、良質に親個体の情報を子個体へと継承できる^[98]。実数値 GA には交叉手法が数種類存在するが、SPX はスケール問題にも対応でき、変数間の依存関係のあるなしに関わらず、良好な結果が得られる。

EDSA では、実問題を取り扱う上で、変数間の依存関係がある問題とそうでない問題、スケールについても対象とする実問題によりパラメータが異なるため、どの問題にも対応できるようこの交叉法を選択した。また最適化問題の n 変数を 1 変数と評価

値の2次元プロットして置き換えているため、変数ごとに極値を持つ個体が判別される。つまり極値となった変数値はそのまま、それ以外の変数値を近傍操作できる。

SPX は対象とする問題が R^n のとき、 $n+1$ 個の親個体 $\vec{P}x_0, \dots, \vec{P}x_n$ を親集団からランダムに選択する。親個体から重心位置 G を求める。

$$\vec{G} = \frac{1}{n+1} \sum_{i=1}^k \vec{P}x_i \quad (4.7)$$

重心 G から拡張率 ϵ をもとに、次世代個体の形成範囲を定め、一様乱数にて次世代個体を生成する。

$$\vec{p}_0 = \vec{G} + \epsilon(\vec{P}x_0 - \vec{G}) \quad (4.8)$$

$$\vec{c}_0 = \vec{0} \quad (4.9)$$

$$\vec{p}_j = \vec{G} + \epsilon(\vec{P}x_j - \vec{G}) \quad (4.10)$$

$$\vec{c}_j = r_{j-1}(\vec{p}_{j-1} - \vec{p}_j + \vec{c}_{j-1}), (j = 1, \dots, n) \quad (4.11)$$

また r_{j-1} は区間 $[0, 1]$ 内の一様乱数 $u((0, 1))$ から

$$r_{j-1} = (u(0, 1))^{\frac{1}{j+1}} \quad (4.12)$$

によって算出される。次世代個体 Cx は

$$\vec{C} = \vec{x}_n + \vec{C}_n \quad (4.13)$$

1 世代当たりの個体数 N と CFD 最適化問題の次数 k の関係が、 $N > k$ であるときは、子個体の数が N に達するまで親個体の選択と子個体の生成を繰り返す。また、子個体が生成された場合、制約条件を満たしていない個体やペナルティリストと一致する個体があれば、生成された個体を保存せず再度生成を行う。

4.2.5 極値近傍探索アルゴリズムの流れ

EDSA では、1 世代毎に個体について評価値と変数の関係から近似曲線と極値を求め、一定の条件に従って交叉候補の個体を絞っていく。交叉には実数値をそのまま使用するシンプレックス交叉を用い、子個体を生成する。その際には子個体が制約条件に沿っているかどうかを判断し、1 世代分の子個体が揃うまで、候補の選択と交叉を繰り返す。そして、その一連の流れを終了世代を迎える、もしくは最適解を導出したと判断するまで繰り返す。終了世代は 60 世代、解の向上が見られなくなった時点から 20 世代解析した後、さらなる向上が見られない場合を終了と判断する。

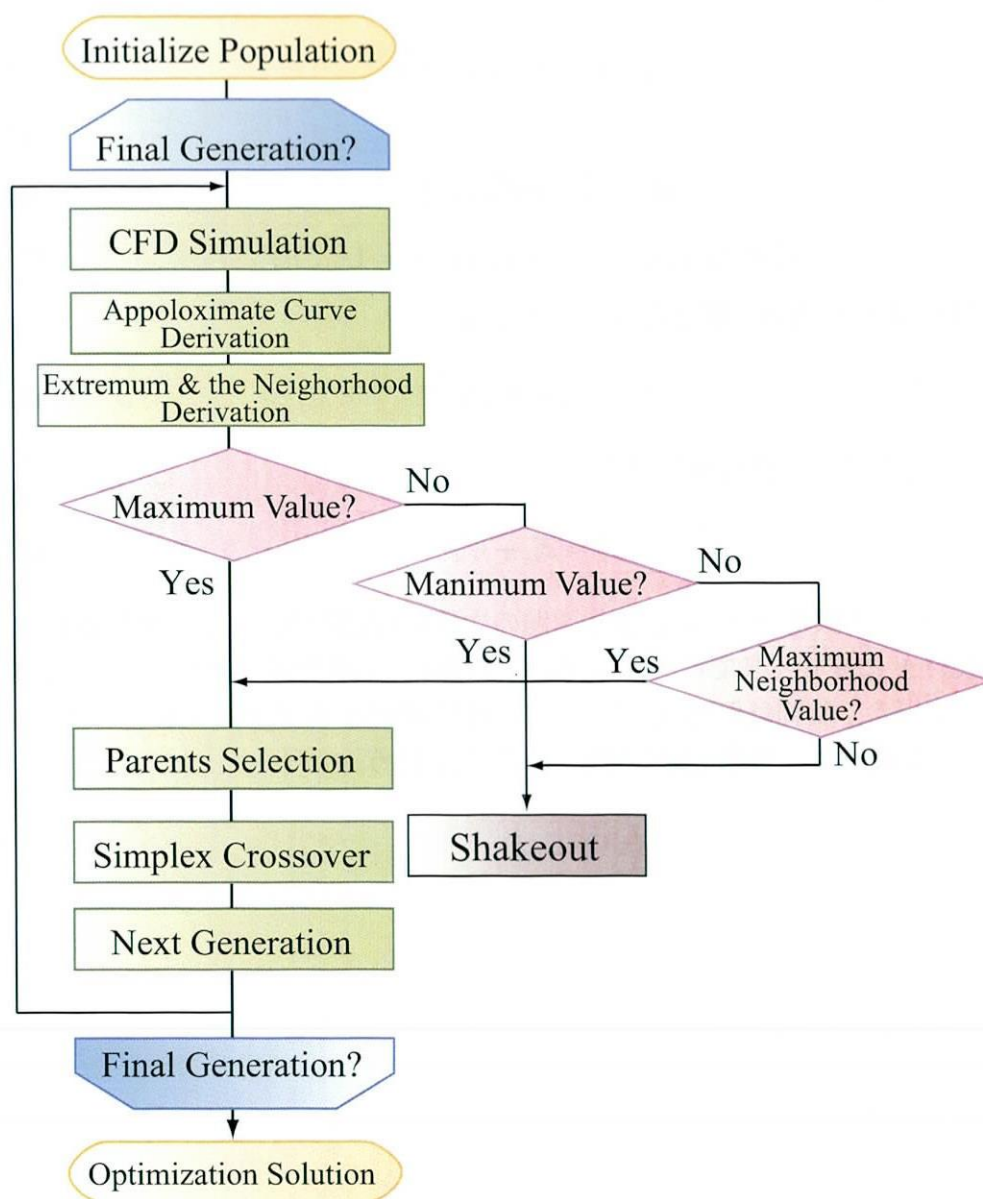


Fig. 4.5 Flow chart of the EDSA

4.3 シミュレーション結果

4.3.1 シミュレーション結果

Fig. 4.6 に遺伝的アルゴリズムと EDSA の収束性能を示す。また、算出された最適解を Table 4.1 に、それぞれの最適速度入力を Fig. 4.7, Fig. 4.8 に示す。また CFD シミュレーションの結果をそれぞれ Fig. 4.9, 4.10 に示す。なお、初期個体は同じものを用いている。

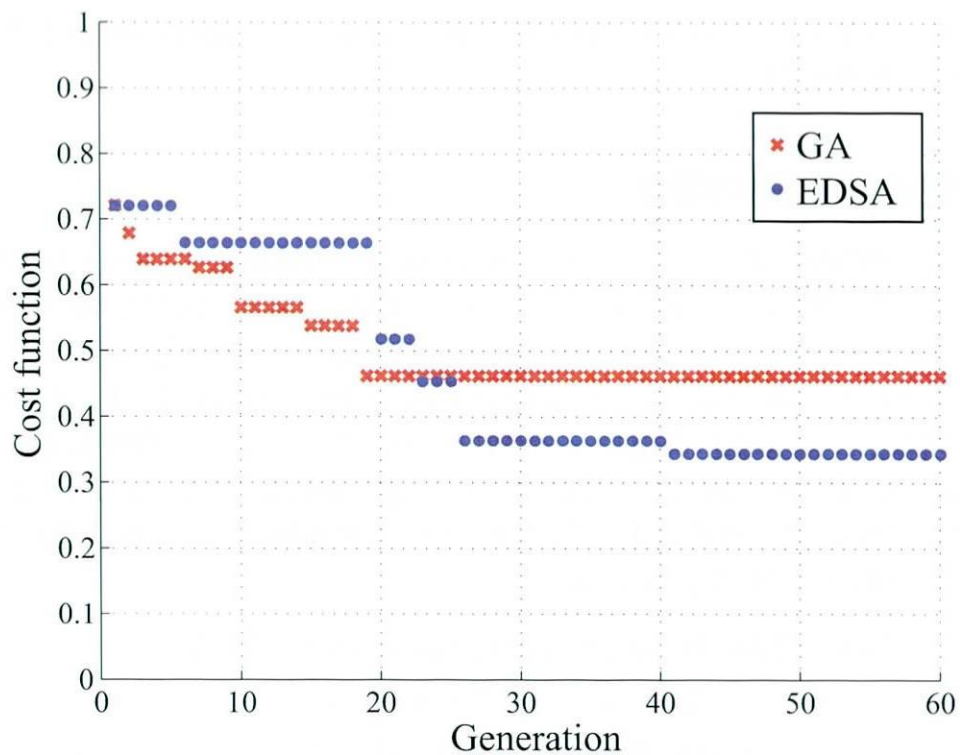


Fig. 4.6 The cost function of the GA and EDSA

Table 4.1 Optimization results of the GA and EDSA

Parameter	GA	EDSA
Cost function J	0.4622	0.3441
Air entrainment A	0.2737	0.1682
Filling time [s]	1.87	1.76
Convergence generation	19	42

Fig. 4.6 の結果より、遺伝的アルゴリズムでは早い段階で良好な解へと到達し順調な最適化が進み、19 世代目で収束をしている。しかしながら評価関数を比較すると、EDSA で導出された評価関数は遺伝的アルゴリズムに比べ、さらに良好な評価をもつ解を見つけている。このことから遺伝的アルゴリズムが局所解に陥っていると推測でき、探索点および繰り返し計算回数の少ない状態で遺伝的アルゴリズムを用いる場合、非常に局所解へと陥り易くなることがわかる。

対して EDSA では、収束には時間がかかっているものの、遺伝的アルゴリズムと比較して良好な解を導出している。これは極値近傍探索を駆使することで、選定に必要なサンプル数を確保しつつ、最良解近傍を探索することで順調に最適解へと到達したと考えられる。つまり良好な解空間では個体を集中させることで収束性能を向上させることができると考えられる。

4.3.2 評価値が同程度の場合

一方で、評価関数が非常に近いときのシミュレーション結果を示す。Fig. 4.11 に遺伝的アルゴリズムと EDSA の評価関数を示し、Table 4.2 には算出された最適解を、そして基まつた最適速度入力をそれぞれ Fig. 4.7, Fig. 4.8 に示す。さらに射出時における CFD シミュレーションの結果についてそれぞれ Fig. 4.14, Fig. 4.15 に示す。Fig. 4.11 の結果から、両者とも早い段階で最適解を導出するに至っている。また最終的に求まつた解の評価値はほぼ同等であることがわかる。なお初期個体は、このシミュレーションにおいても EDSA、遺伝的アルゴリズムともに同じものを用いているが、前節とは異なる新たに生成したものを用いている。

Table 4.2 Another optimization results of GA and EDSA

Parameter	GA	EDSA
Cost function J	2.2503	2.2197
Air entrainment A	0.7768	0.4126
Filling time [s]	1.0851	1.6008
Convergence generation	23	26

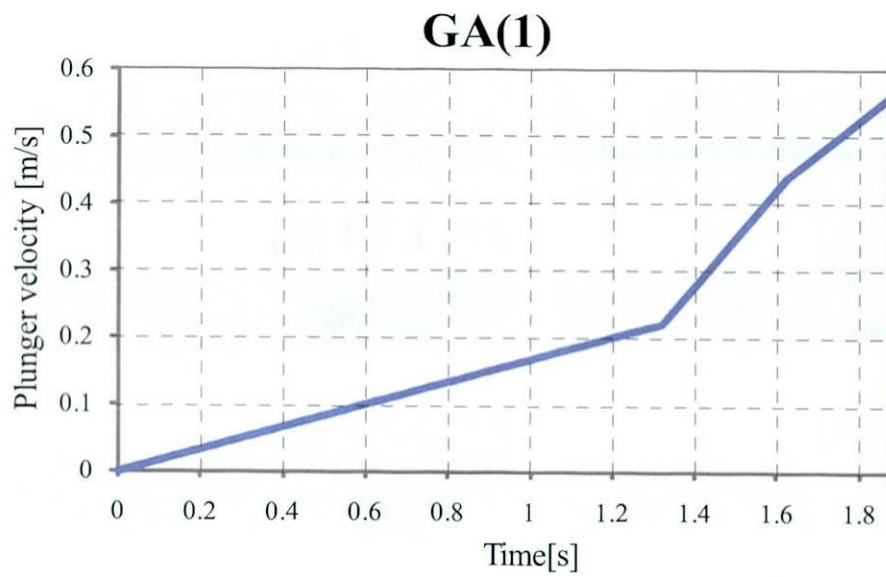


Fig. 4.7 Optimum plunger velocity input of GA

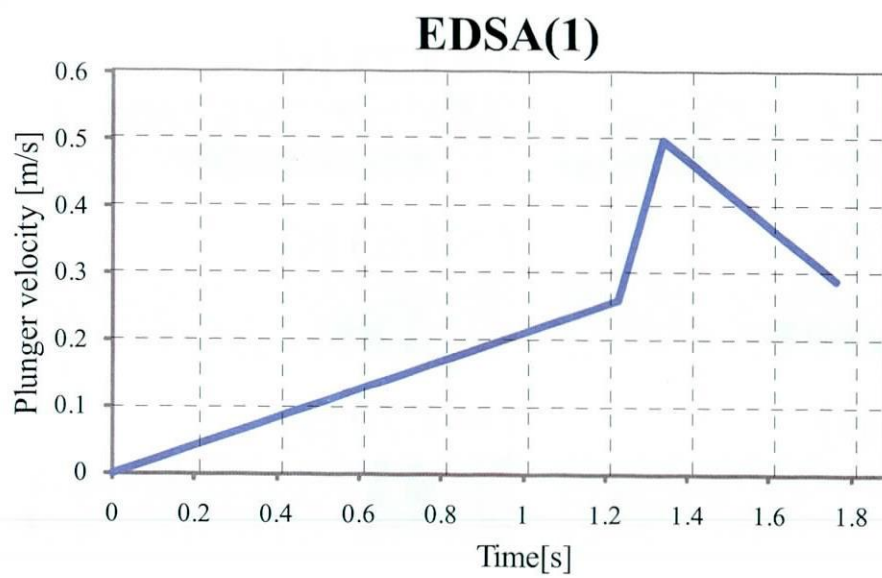


Fig. 4.8 Optimum plunger velocity input of EDSA

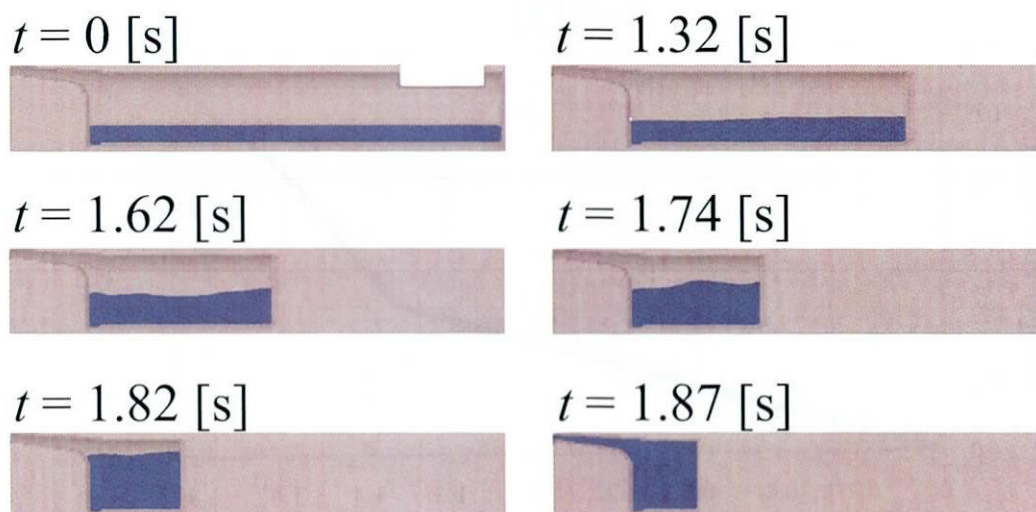


Fig. 4.9 Optimum plunger velocity of GA

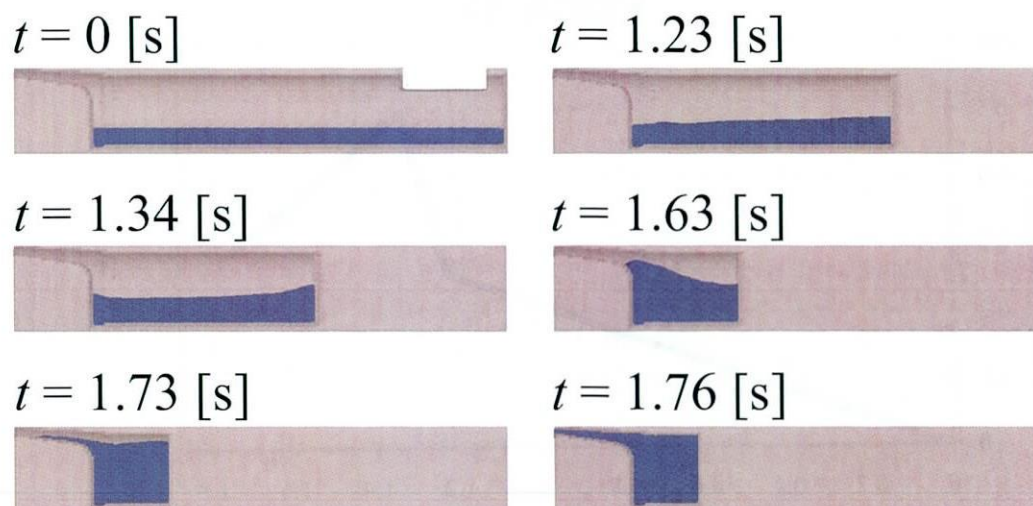


Fig. 4.10 Optimum plunger velocity of EDSA

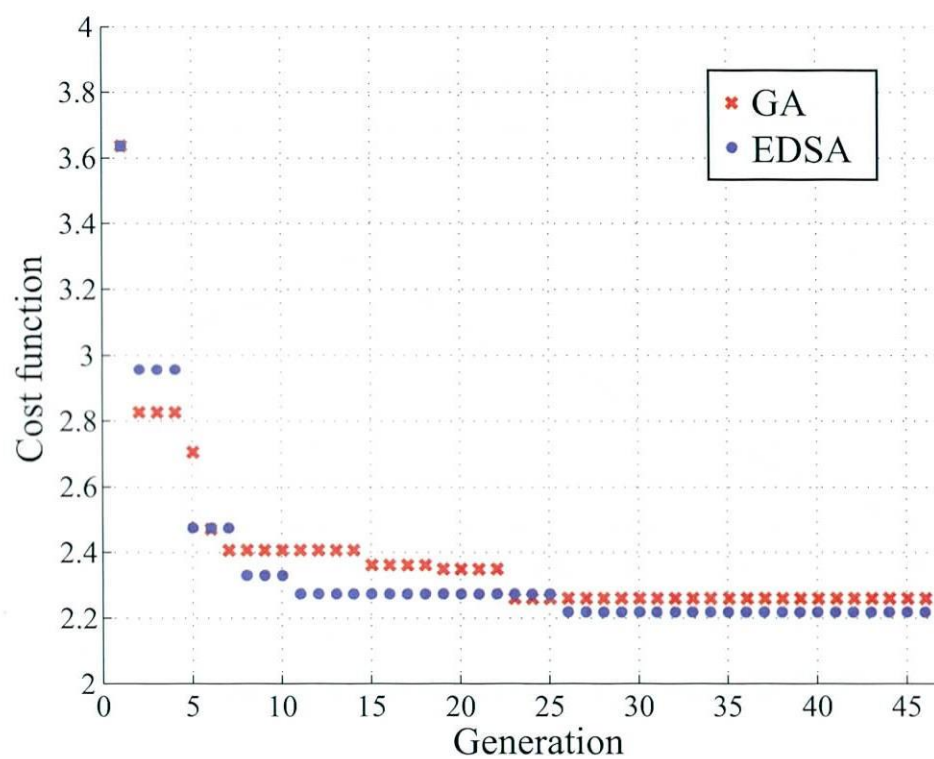


Fig. 4.11 Another result of cost function of GA and EDSA

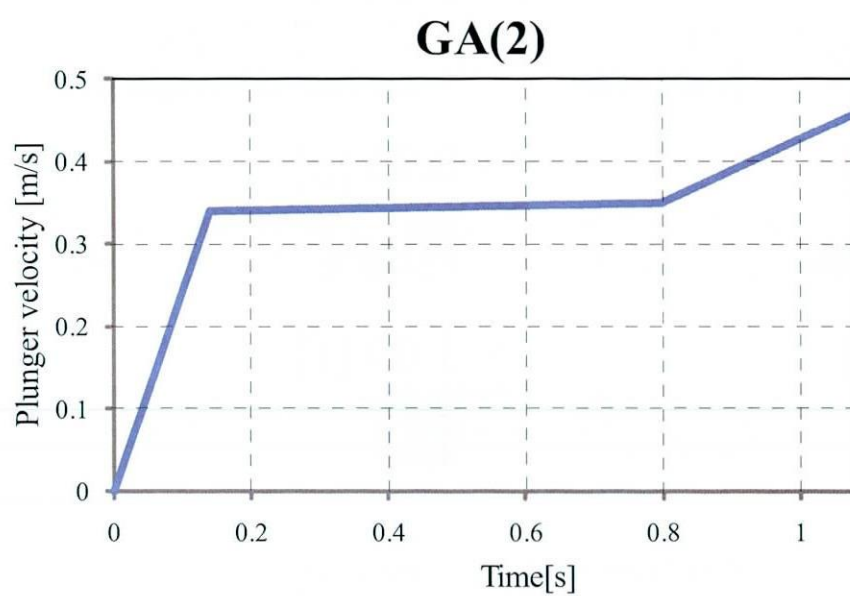


Fig. 4.12 Optimum plunger velocity input of GA

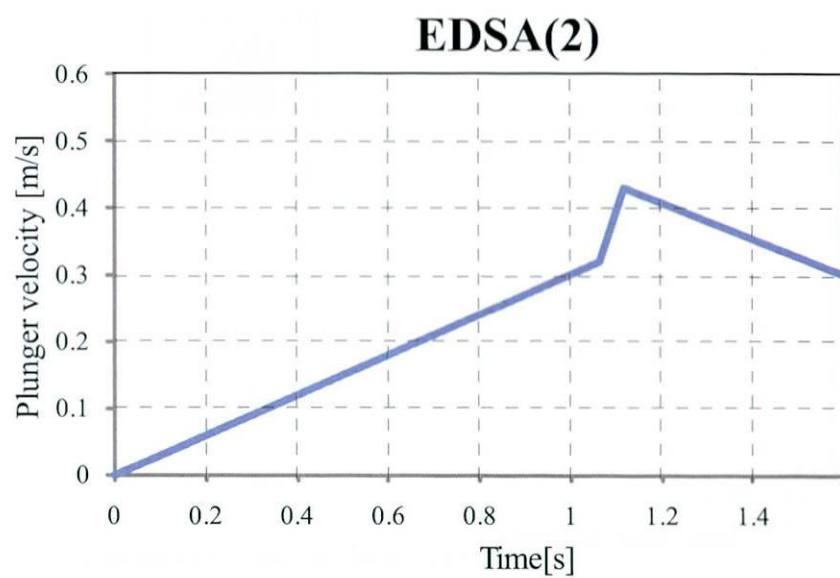


Fig. 4.13 Optimum plunger velocity input of EDSA

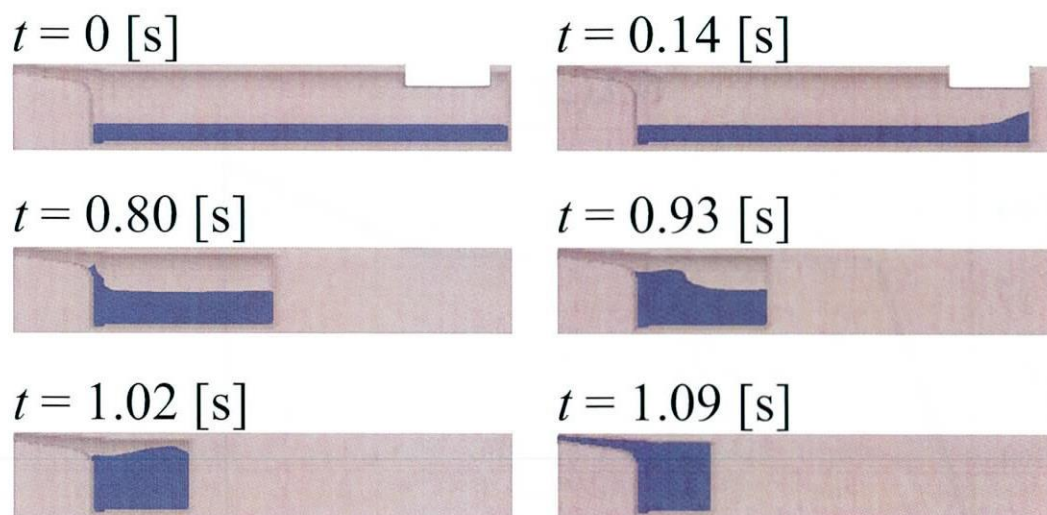


Fig. 4.14 Optimum plunger velocity of GA

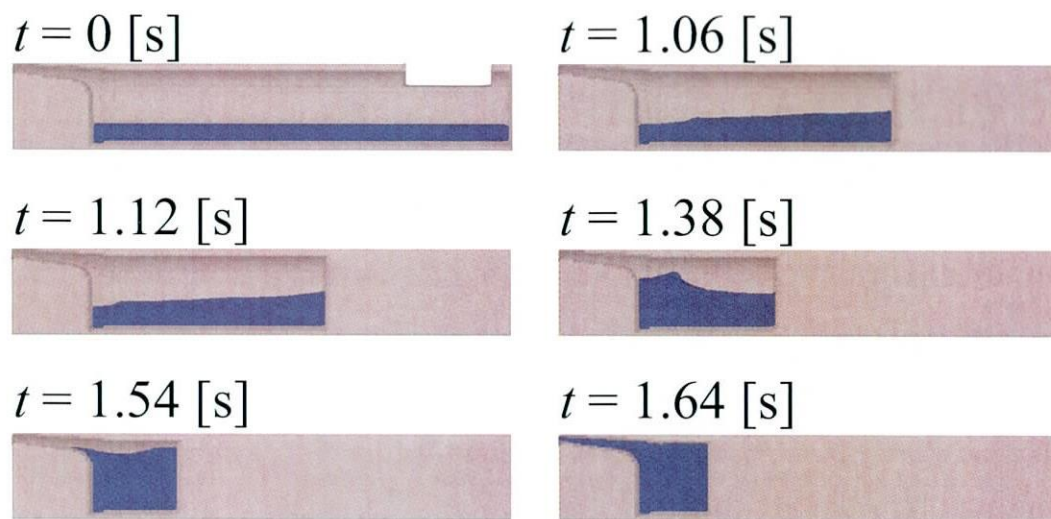


Fig. 4.15 Optimum plunger velocity of EDSA

4.4 実験結果

Table 4.3 に示す各アルゴリズムの算出された解を用いたときのブリスタ試験結果を Fig. 4.16, Fig. 4.17 に示す. 気泡の面積は $0.005[\text{m}]$ 方眼紙を用い, 直径 $0.001[\text{m}]$ 以上の気泡があるマス数をカウントすることにより求めた.

Fig. 4.16, Fig. 4.17 より, シミュレーション結果と同様に, EDSA は遺伝的アルゴリズムに比べ空気巻き込み量の少ない良好な製品を作ることができた.

もう一方で, 評価値がほぼ同等な場合の結果を Fig. 4.18, Fig. 4.19 に示す. またこのときの速度入力を Table 4.4 に示す. これらの結果より, 評価値は同程度であっても実験における空気巻き込み量は大きく異なる結果となった. したがって, EDSA を用いることで実験でも良好な良化を得ることのできる解を求めることができることがわかる.

Table 4.3 Plunger velocity parameters

GA	Time[s]	Velocity[m/s]	Position[m]
$i = 1$	1.32	0.220	0.145
$i = 2$	1.62	0.440	0.245
$i = 3$	1.87	0.560	0.367
Cost function 0.4622			
Total area of the air bubble : $1.267 \times 10^{-3}[\text{m}^2]$			
EDSA	Time[s]	Velocity[m/s]	Position[m]
$i = 1$	1.23	0.26	0.160
$i = 2$	1.34	0.50	0.200
$i = 3$	1.76	0.29	0.367
Cost function 0.3441			
Total area of the air bubble : $0.845 \times 10^{-3}[\text{m}^2]$			

Table 4.4 Plunger velocity parameters of the another optimization results

GA	Time[s]	Velocity[m/s]	Position[m]
$i = 1$	0.14	0.34	0.024
$i = 2$	0.80	0.35	0.250
$i = 3$	1.09	0.46	0.367
Cost function 2.2503			
Total area of the air bubble : $1.881 \times 10^{-3}[\text{m}^2]$			
EDSA	Time[s]	Velocity[m/s]	Position[m]
$i = 1$	1.06	0.32	0.170
$i = 2$	1.12	0.43	0.190
$i = 3$	1.60	0.30	0.367
Cost function 2.2197			
Total area of the air bubble : $1.157 \times 10^{-3}[\text{m}^2]$			

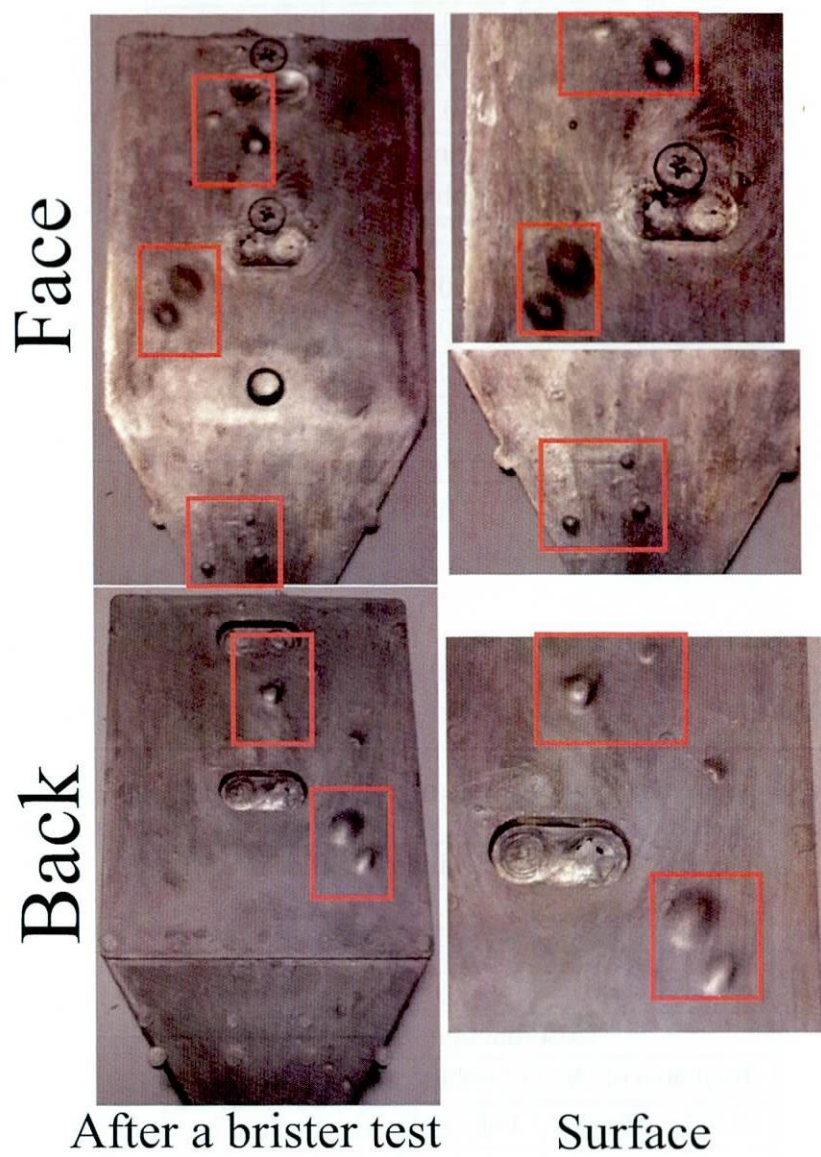


Fig. 4.16 Result of brister test of GA

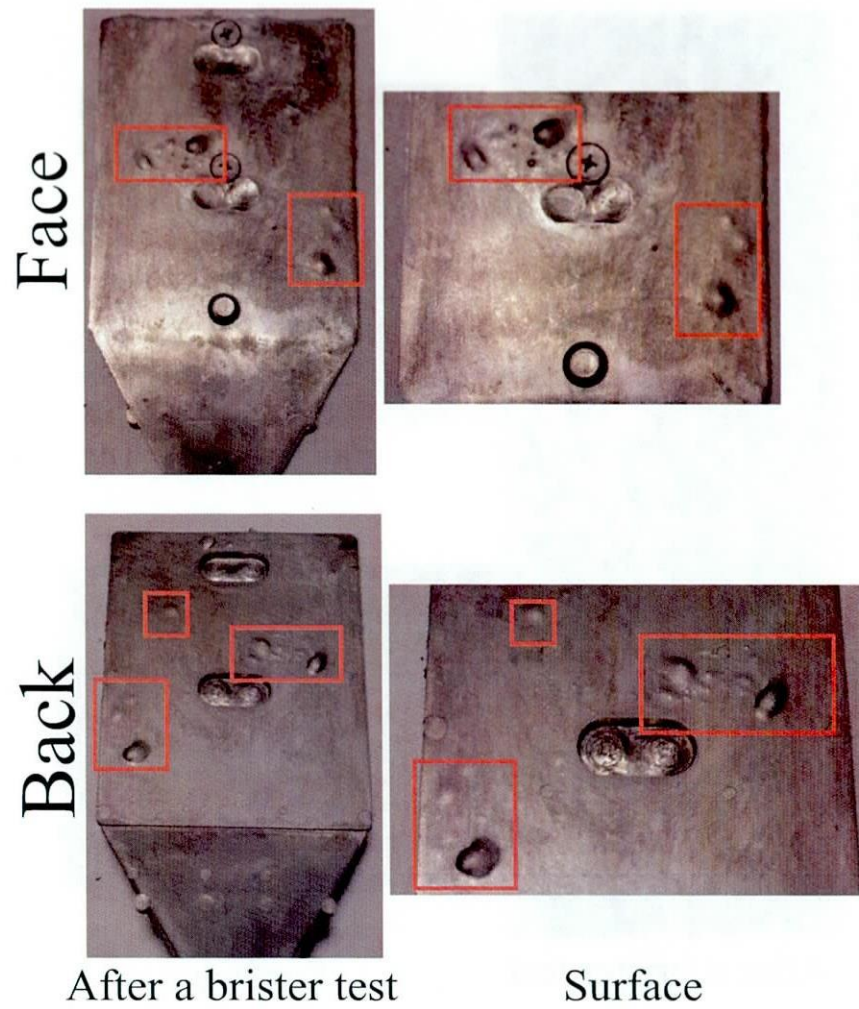


Fig. 4.17 Result of brister test of EDSA

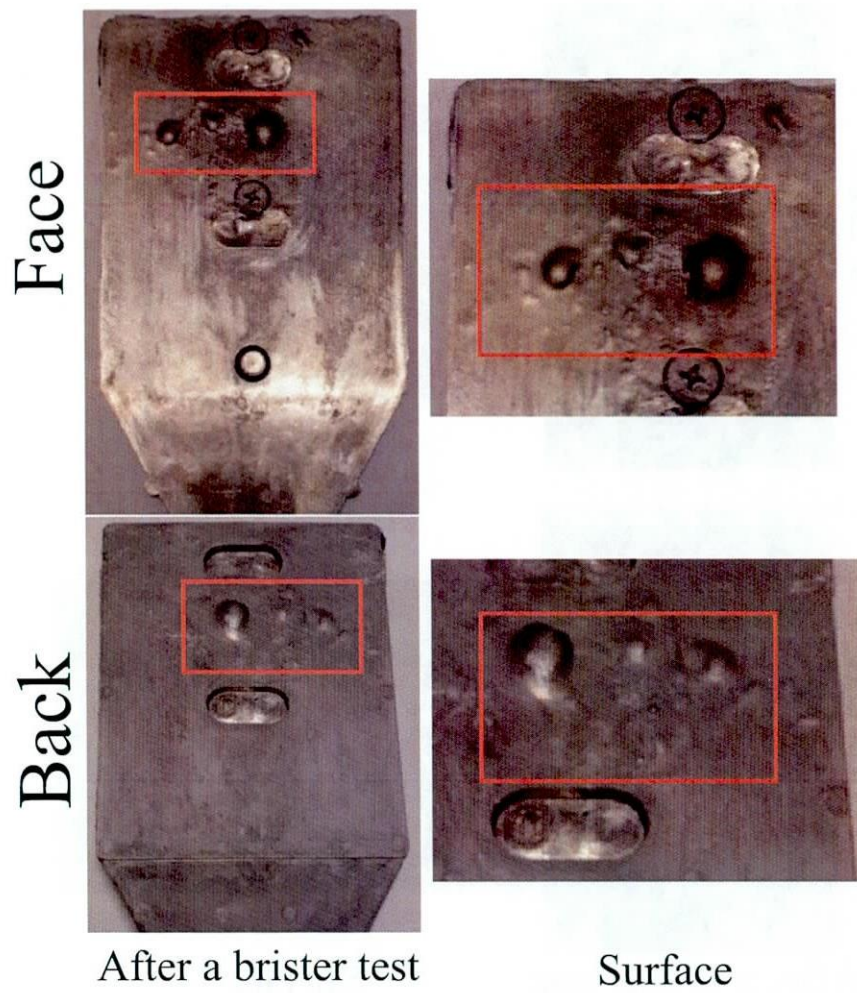


Fig. 4.18 Another result of brister test for GA

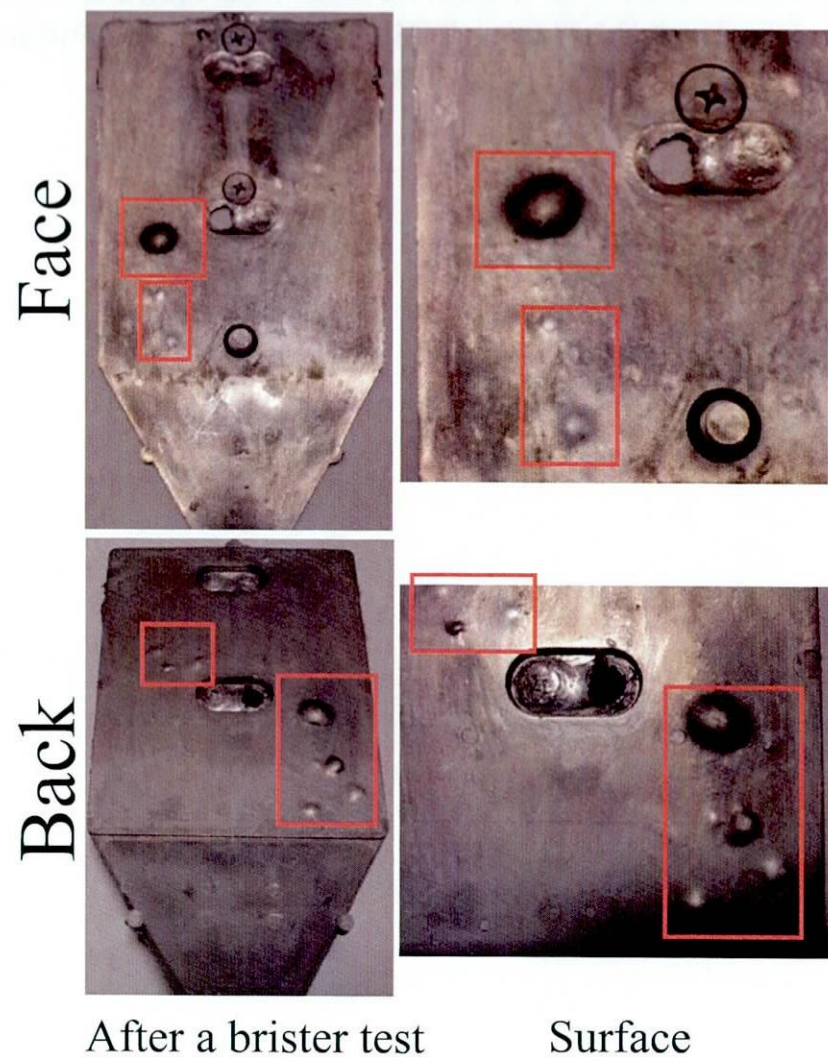


Fig. 4.19 Another result of brister test for EDSA

4.5 まとめ

本章では実験において、良好な結果が得ることのできる解の算出を目的とした極値近傍探索アルゴリズム (EDSA) を提案した。具体的には極値判別と近似曲線を用いて、大域的に良好となっている解を算出するアルゴリズムである。結果、遺伝的アルゴリズムに比べ、実験において良好な結果を得られることがわかり、また評価値が同等であっても、実験では大きな差が生まれ、EDSA では良好な実験結果を得られることがわかった。さらに解空間における解の向上が見込まれる空間を予測、集中的に探索することで収束性能を向上させることが期待できることがわかった。

第5章 多極集中型最適化アルゴリズム

5.1 概要

前章の結果より，個体の配置が収束性能に大きく影響を与え，そして解の向上が見られる空間では集中的に探索点を配置させた方が収束性能が向上することが予想された．そこで本研究では，各探索点（前章までの個体を意味する）が他の位置や評価を考慮し，自ら配置を決めていく最適化アルゴリズムを構築する．具体的には，初期の探索点配置問題に至っては分散制御則によって探索点を解空間全体に分散させ，次世代の探索点配置問題に至っては集中アルゴリズムを構築し，自らの近辺に情報があれば，その情報から解の向上が見込まれる領域へと移動する法則を導入することで良好な解空間を集中的に解析するようにし，解の収束性能の向上を図る．提案する多極集中型最適化アルゴリズム (Multi-subcenters Solution Search Algorithm: 以下 MSSA と表記) の流れを Fig. 5.1 に示す．本手法の有効性は，ダイカストにおけるプランジャ速度の最適化に適用して，実験結果を通して示す．

5.2 ランダム生成による探索点の初期配置

探索点の初期配置は収束性能に大きな影響を与えるだけでなく，大域的な最適解を導出するうえで，未解析領域の発生は局所解を発生させる大きな原因と考えられる．特に遺伝的アルゴリズムなどは局所解への陥りを回避するため，突然変異率という概念を用いて未解析領域への探索点の配置を行っている．しかしながら突然変異率を上げることで収束性能が低下するため，今回ような最適化の場合，計算時間との兼ね合いからその確立が低く設定されている．

一般的に個体の配置はランダム生成で行う．これは複雑な解空間においては均等配置を計算によって求めることが難しいこと，そして探索点数が多いためである．なお，ランダム生成法としては疑似乱数生成器の一つであるメルセンヌ・ツイスター法^[99]が広く用いられており，本研究室でもそれを適用している．

Fig. 5.2 は，メルセンヌ・ツイスターのランダム生成を用いて探索点が配置された様子を示している．このとき領域の端は境界制約を示し，グラフ内部の実線は特定の制約条件の関数 $f(q)$ の境界を示している．図より斜線で示した箇所には探索点が配置されておらず，このことからランダム生成だけでは探索点の配置に偏りができ未解析領域が発生しやすくなることが予想される．

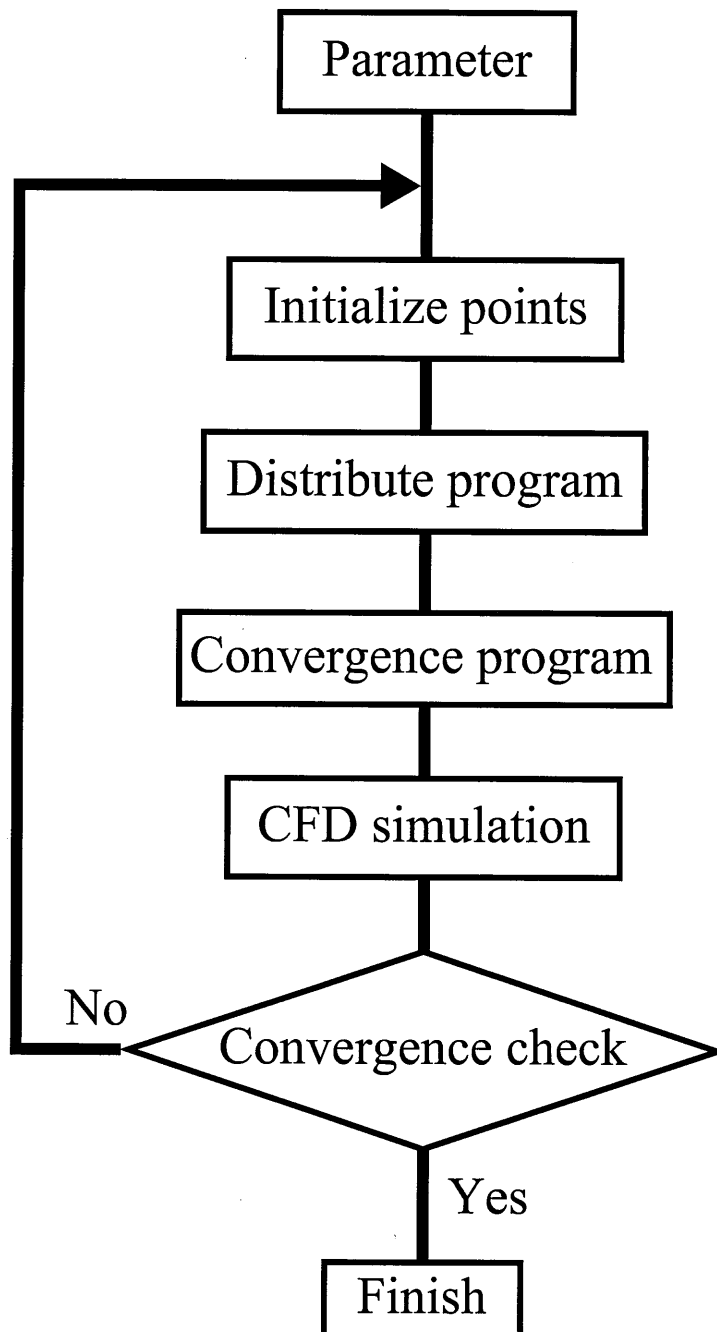


Fig. 5.1 Flowchart of the Multi-subcenters Solution Search Algorithm

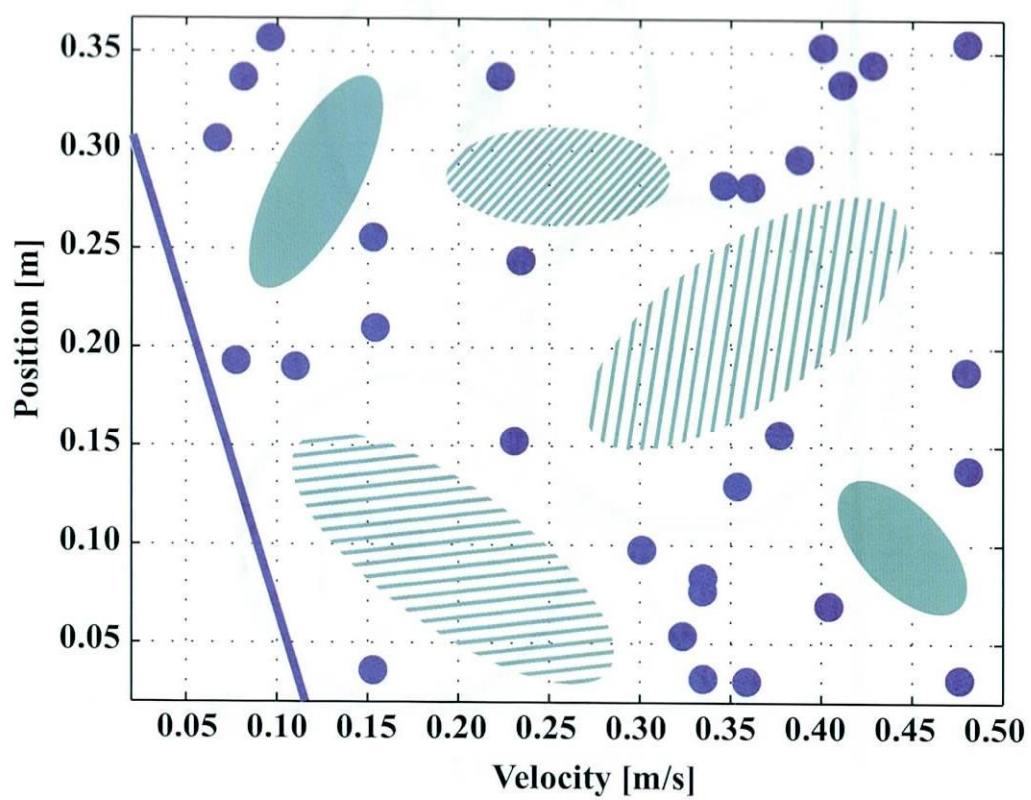


Fig. 5.2 Generated agents by using random

5.3 多極集中型最適化アルゴリズムの設計

5.3.1 分散制御アルゴリズムの設計

解空間上に探索点を均等に配置するために行うため、分散制御を用いる。この分散制御側は探索点自身が他の探索点の位置を把握して、自身の位置を決定するものである。制御側としては被覆制御^[100]の考え方をに基づいている。分散制御側の概念図を Fig. 5.3 に示す。

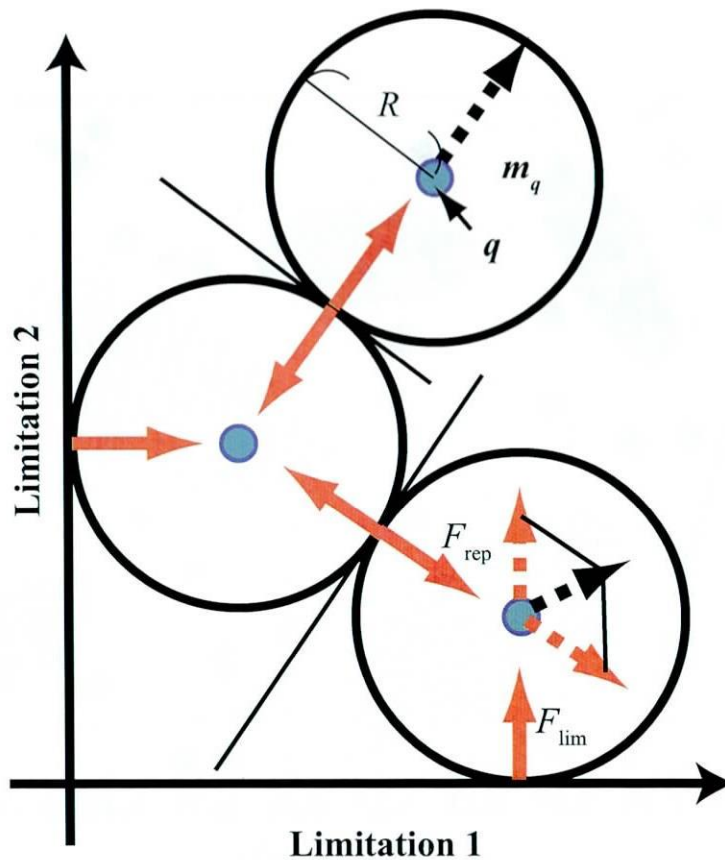


Fig. 5.3 Basic concept of distribute control

まず探索点同士の間隔を調整するため、各エージェントに円を持たせ、その円同士の衝突によって探索点 q の配置を行う。初期状態の円の半径 R は 0 とし、移動の有無で Eq.(5.1) に示すように膨張させていく。ここで t はサイクルを示し、 R_{add} は加算する円の半径、 m_q は探索点 q の移動量を示す。移動量は Eq.(5.2) に示すように探索点の位置ベクトル $q = (q_1, q_2, \dots, q_i, \dots, q_n)$ と、その他の探索点の位置ベクトル $p = (p_1, p_2, \dots, p_i, \dots, p_n)$ との距離が $2R$ 以下となったとき、それらの距離に応じた斥力 F_{rep} が働くようにする。ここで n は求めたい変数の数を示している。これらの斥力を合成をし、Eq.(5.3) を用いて移動ベクトル m_q とする。

$$R(t+1) = \begin{cases} \forall R(t) + R_{\text{add}}, & \text{if } \forall \mathbf{m}_q = 0 \\ \forall R(t), & \text{otherwise} \end{cases} \quad (5.1)$$

$$F_{\text{rep}} = \frac{1}{\|\mathbf{q} - \mathbf{p}\|}, \quad (\mathbf{q} \neq \mathbf{p}) \quad (5.2)$$

$$\mathbf{m}_q = \begin{cases} \frac{(\mathbf{q} - \mathbf{p})}{F_{\text{rep}}^2}, & \text{if } \|\mathbf{q} - \mathbf{p}\| \leq 2R \\ 0, & \text{otherwise} \end{cases} \quad (5.3)$$

円がそれぞれの変数に対する境界条件 $B_{\text{max},k}, B_{\text{min},k}$ に触れた場合は Eq.(5.4) に示す移動量 F_{lim} が起こる. ここで q_k は探索点の各変数の値を示している.

$$\mathbf{m}_q = \begin{cases} -F_{\text{lim}}, & \text{if } q_k - R \leq B_{\text{min},k} \\ F_{\text{lim}}, & \text{if } q_k + R \geq B_{\text{max},k} \\ 0, & \text{otherwise} \end{cases} \quad (5.4)$$

($1 \leq k \leq n$)

また特定の制約条件の関数 $f(\mathbf{q})$ に対して, 移動後の探索点 $\mathbf{q}(t) + \sum \mathbf{m}_q$ がその制約 const に触れた場合は斥力に対する移動は無くなる.

$$\mathbf{q}(t+1) = \begin{cases} \mathbf{q}(t), & \forall f(\mathbf{q}(t) + \sum \mathbf{m}_q) \notin \text{const} \\ \mathbf{q}(t) + \sum \mathbf{m}_q, & \text{otherwise} \end{cases} \quad (5.5)$$

Eq.(5.1) ~ Eq.(5.5) の動作を探索点の移動が無くなるまで繰り返すことで分散させる. このように探索点の間隔を風船のように考え, 移動がなくなるまで円を膨らまし続けることで解空間全体に探索点を配置できる. また MSSA はすべてベクトルで構築されているため多変数への対応が可能となる.

5.3.2 集中アルゴリズムの設計

効率良く解の探索を行うためには解の向上がみられる領域を集中的に探索することが望ましい. そこで各探索点に近辺の評価情報を考慮し, 解の向上が見込まれる領域へと移動するアルゴリズムを構築し適用する. Fig. 5.4 は次世代探索点の移動方法を示す. はじめは探索点を分散制御を使って解空間全体に配置を行う. 広がった探索点 \mathbf{q} は分散時に用いた半径の最大半径 R_{max} の領域内にある解析済みの探索点 \times を参考にして, 解の向上が見込まれる領域へと移動していく.

Fig. 5.5 に探索点の移動の概念図を示す. ここで \mathbf{g} は \mathbf{b}_i で構成される点の重心, c_i は荷重, \mathbf{g}_u は荷重を与えた後の重心, \mathbf{u} は重心 \mathbf{g} から重心 \mathbf{g}_u までの移動ベクトル, \mathbf{q}_v は探索点 \mathbf{q} から \mathbf{v} 移動した後の探索点を表している. この \mathbf{g} から \mathbf{g}_u への重心移動を傾きとして考え, 近辺にある傾きを調べて \mathbf{q} の移動方向 \mathbf{v} を決定する.

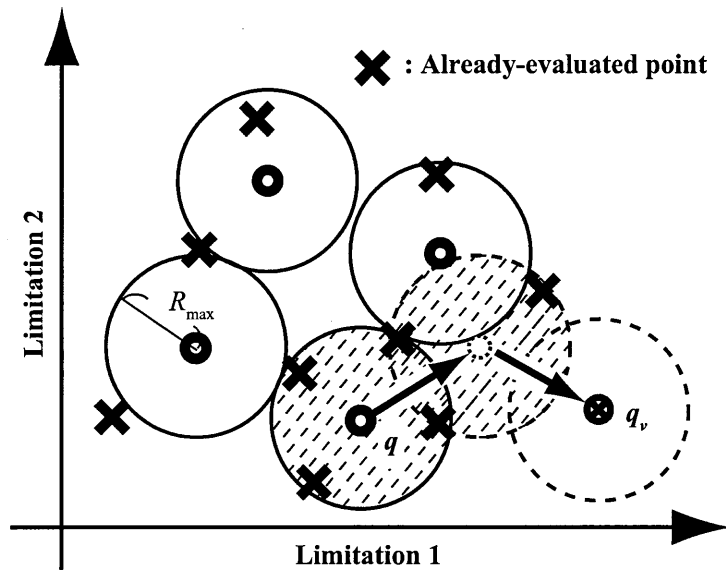


Fig. 5.4 Setting the searching point of next generation

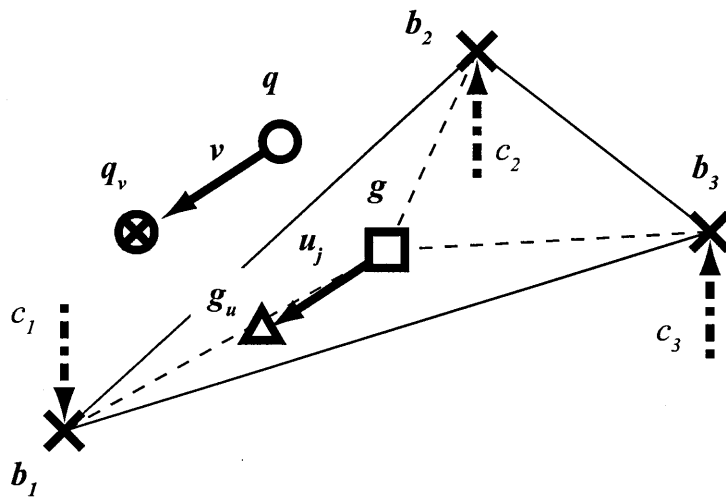


Fig. 5.5 Moving concept of searching point

q_v は次式のようにあらわす.

$$q_v = q + v \quad (5.6)$$

近辺の定義は分散の時に利用した R_{\max} の範囲内として考える. 傾きを生成するに当たり, 解析済みの探索点の必要個数は $n+1$ 以上なければならない. これは変数が1つの時, 変数と評価値の2次元のグラフから傾きを算出する場合は2つの点が必要であり, また3次元の時は3つ必要であるからである. つまり求めたい変数の数 n に対して $n+1$ 個の解析済みの個体を要する. また $n+1$ 個よりも多く解析済みの探索点が存

在する場合は次式のように組み合わせを考えて生成する．ここで a は R_{\max} の領域内に
ある解析済みの探索点の数を示す．

$$d = \begin{cases} aC_{n+1} & (a \geq n+1) \\ 0 & (a < n+1) \end{cases} \quad (5.7)$$

q の移動方向である v は周辺の d 個の傾きベクトル u_j を平均して得る．ただし $d = 0$ の場合もしくは移動後が制約条件 (式中は const と略記) を満たしていない場合、移動距離は 0 とする．

$$v = \begin{cases} 0 & (d = 0 \text{ or } f(q + v) \notin \text{const}) \\ \frac{\sum_{j=1}^d u_j}{d} & \text{otherwise} \end{cases} \quad (5.8)$$

また傾きベクトル u_j は、 $n+1$ 個の評価済みの探索点 b_i で構成される重心ベクトル g から、評価 c_i を荷重として考えたときの重心ベクトル g_u までの移動方向とする．

$$u_j = g_u - g \quad (5.9)$$

ここで重心 g は次式で表し

$$g = \frac{\sum_{i=1}^{n+1} b_i}{n+1} \quad (5.10)$$

また重心 g_u は次式で表す．

$$g_u = \frac{\sum_{i=1}^{n+1} b_i \frac{1}{c_i}}{\sum_{i=1}^{n+1} \frac{1}{c_i}} \quad (5.11)$$

これらの法則を設定することで各探索点は近辺 (R_{\max} の範囲内) に評価値が規定数 $n+1$ 以上あった場合、評価情報を頼りに解の向上が見込まれる領域へと移動を行い、結果、探索点を集中させることができる．

さらに集中性を高めるため、引力アルゴリズムを加える．Fig.5.6 は引力アルゴリズムの概念図を示す．探索点は集中アルゴリズムの移動則によって移動を行う．その移動途中で、良好な解析点近傍を通過した場合、引力アルゴリズムが働き、移動ベクトルの向きをその引力中心へと向ける．

式 (5.12) は引力アルゴリズムの基礎式を示す．ここで x_g は引力中心、 X は解析済み探索点の集合、 b は解析済みの探索点、 R_{\max} は分散時に掛かった最大半径、 w_R はユーザーが決める有効半径の重み、 $w_R R_{\max}$ は引力の有効半径を示す．また世代の増加に伴い多くの探索点を良好な解空間に集中させるため、 k を現在の世代数として $w_R R_{\max}$ に積算する．さらに、引力点を厳選するため b は式 (5.13) に満たすものとする．ここで、 \bar{B} は全ての評価点の平均を示し、 b_a はユーザーが決定する閾値である．

$$x_g = \min \{f(x_g) | x_g \in X : \|x_g - b\| \leq 2w_R R_{\max} k\} \quad (5.12)$$

$$b : b/\bar{B} \leq b_a \quad (5.13)$$

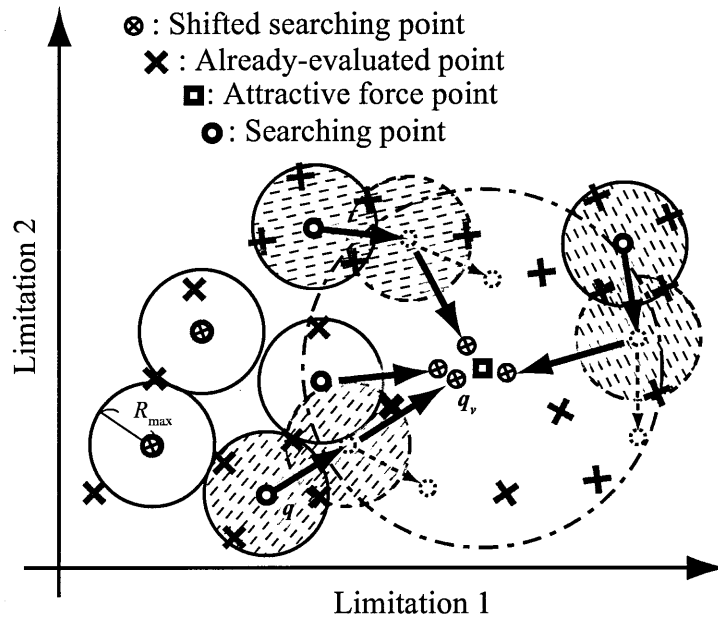


Fig. 5.6 Attractive force algorithm

5.4 2変数における最適化アルゴリズムの収束性能の比較

5.4.1 初期探索点の比較

分散制御則とランダム生成との配置の違いについて比較を行う。探索点の配置の検証を行う。Fig. 5.7はFig. 5.2に分散制御則Eq.(5.1)～Eq.(5.5)を適用したときの物である。Fig. 5.7はFig. 5.2に比べ偏りがなく、十分に分散を行えていることがわかる。なお、この初期配置問題に関して、遺伝的アルゴリズムを用いて収束性能を比較・検証したときの結果を付録Dに示す。

5.4.2 収束性能と探索点配置の比較

MSSAの有効性を検証するため、遺伝的アルゴリズムとMSSAの収束性能を比較する。

MSSAの性能をFig. 5.8の解空間を用いて評価を行う。Fig. 5.8の解空間は実際にCFDシミュレータを用いて速度と位置とその評価値を表した図である。この図は標高が低いほど評価が良くなっていることを示しており、全数探索の結果、大域的な最適解は0.58[m/s], 0.22[m]であることがわかっている。また、その大域的な最適解周辺は凹凸が激しいことから多数の局所解が存在していることがわかる。

収束性能は最大世代数を1000世代として大域的な最適解である0.58[m/s], 0.22[m]を何世代で発見したかで比較する。また、この最適化の試行は10回行い検証する。このとき用いた最適化の各パラメータをTable 5.1とTable 5.2に示し、変数は2変数、各世代の個体は30個体とする。ここで、引力アルゴリズムの変数である w_R と b_a におい

では、今回の問題はシミュレータの評価を直接評価関数に加えているため、計算誤差により解空間は非常に小さい多峰性を多く持つこととなる。そのため探索点を一点に集中させてしまう引力アルゴリズムの影響を抑え、多極性を重視することにした。そこで R_{\max} の範囲を引力にするために $w_R = 0$, $b_a = 0$ とした。引力アルゴリズムを適用した場合の多極性の変化について詳細は付録Dに示す。

Table 5.1 Parameters for GA

Number of variable	2
Number of population	30
Number of elite preservation	2
Mutation fraction	0.03
Crossover fraction	0.80

最適化を行うにあたり、条件を公平にするため、それぞれの試行開始時の個体配置はMSSA、遺伝的アルゴリズムの双方とも同じものを用いる。収束結果をFig. 5.9に示す。図から遺伝的アルゴリズムに比べMSSAは収束性能が良く、また収束平均は7.7世代であり、少ない計算回数で最適値を求めることができていることがわかる。また

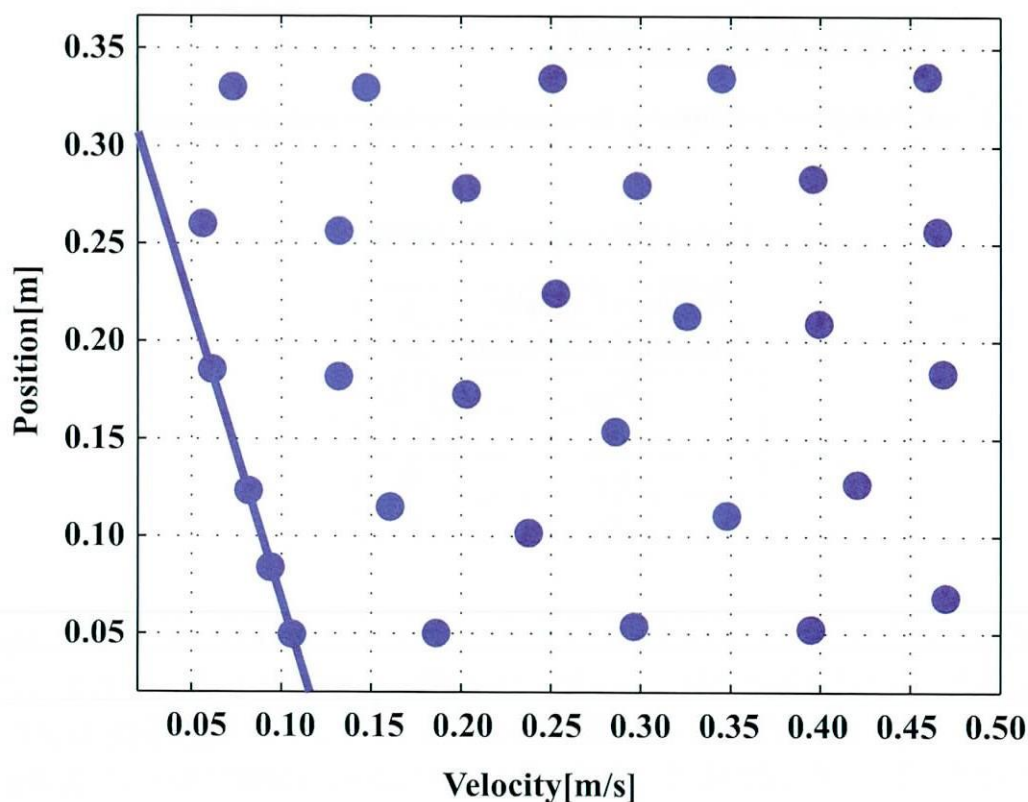


Fig. 5.7 Arranged agents by using distribute control after the random location

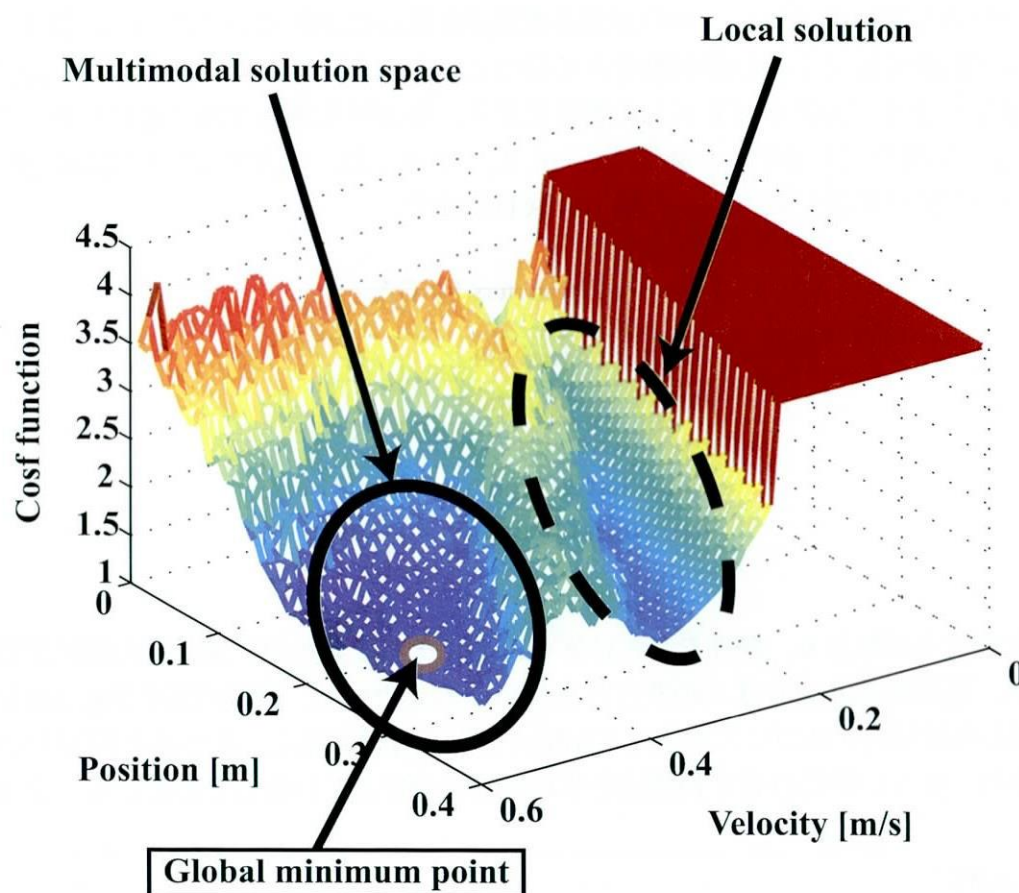


Fig. 5.8 Example of relationship between two variables and air entrainment

Table 5.2 Parameters for MSSA

Number of variable	2
Number of population	30
R_{add}	0.001
F_{lim}	0.1
w_R	0
b_a	0

遺伝的アルゴリズムにおいて2回目と6回目の試行では、最大の1000世代まで計算を行っても最適値へ到達することはなかった。これは探索点が局所解へ陥ってしまっているためである。これらの結果から、MSSAを用いることで大域的な最適解を素早く、また局所解に陥ることなく求めることができることがわかる。5変数における性能調査については付録Eに示す。

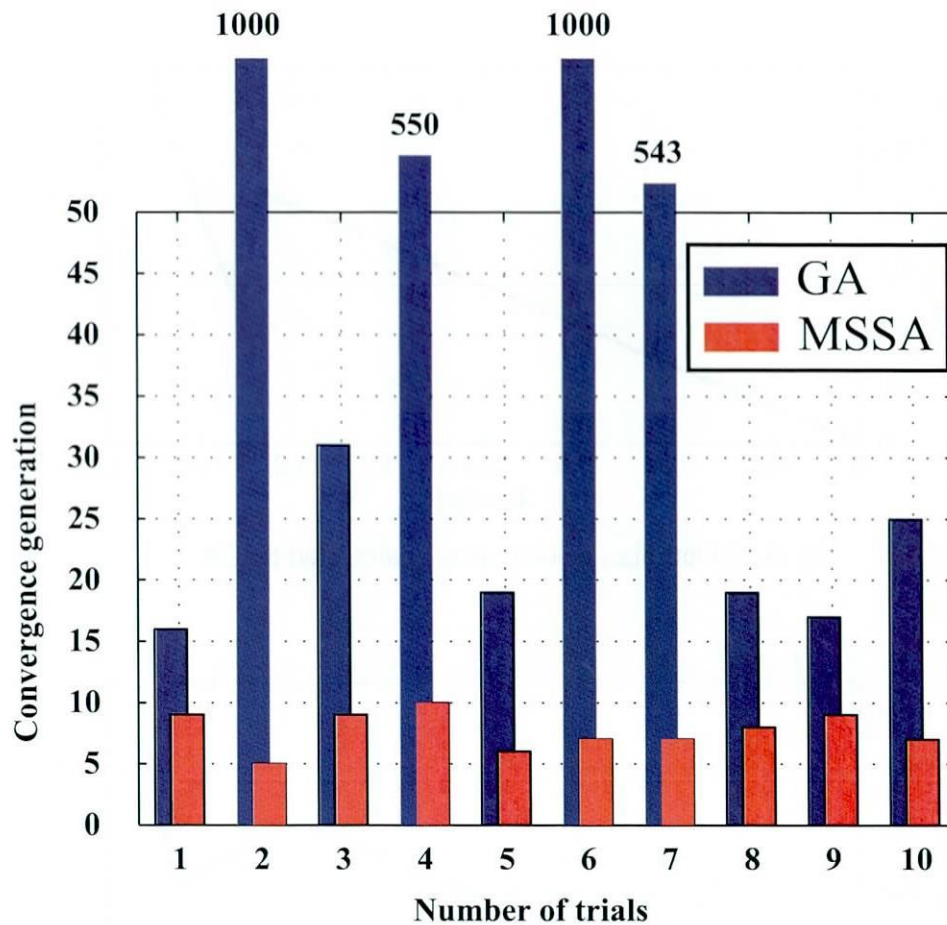


Fig. 5.9 Convergence performance comparison between MSSA and GA

5.5 最適化結果

MSSAの有効性を確かめるため、遺伝的アルゴリズムとMSSAの最適化結果をTable 5.3に示す。ここで条件を公平にするため、初期個体は同じ物を用いている。また各設定パラメータは変数を5として遺伝的アルゴリズム、MSSAそれぞれTable 5.1, Table 5.2と同様の物を用いている。遺伝的アルゴリズムとMSSAで算出された速度カーブをFig. 5.10とFig. 5.11に、そのシミュレーション結果をFig. 5.12とFig. 5.13にそれぞれ示す。シミュレーションから射出時間、評価値がほぼ同じこともあり、ほとんど入力に差異がないことがわかる。

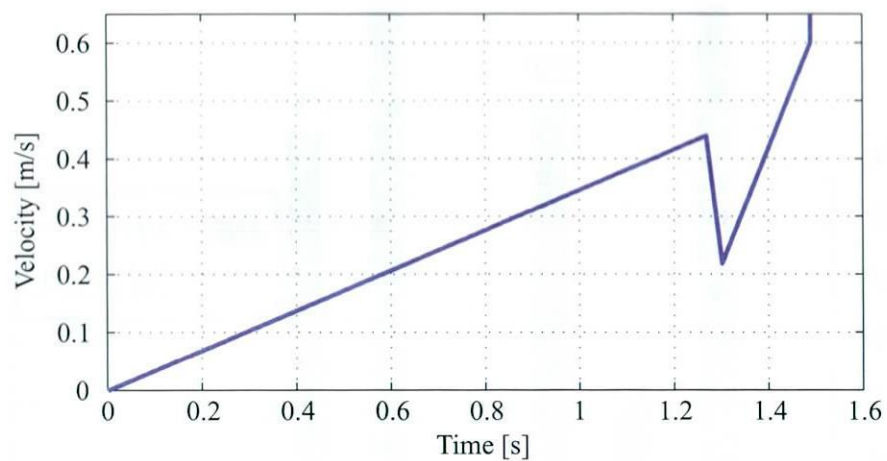


Fig. 5.10 Optimum velocity curve calculated by GA

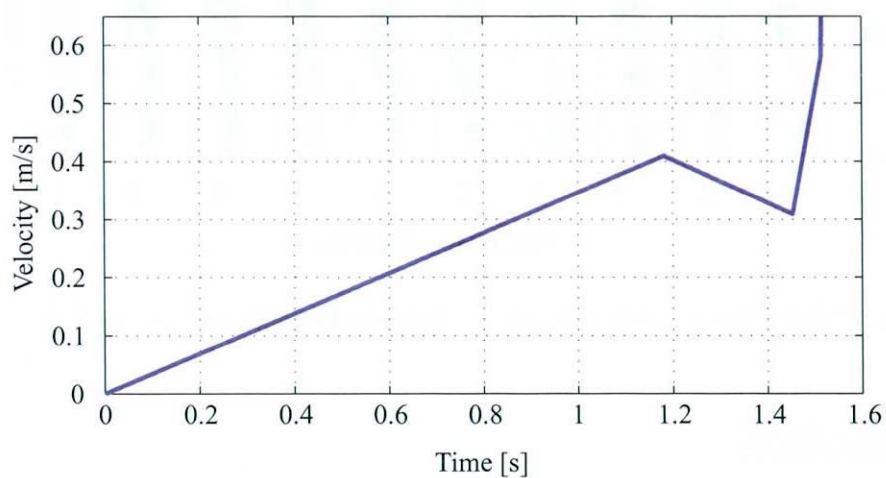


Fig. 5.11 Optimum velocity curve calculated by MSSA

Table 5.3 Performance comparison of optimization results

Parameter	MSSA	GA
Cost function J	1.789	1.831
Air entrainment A	0.193	0.221
Filling time [s]	1.50	1.50
Convergence generation	39	54

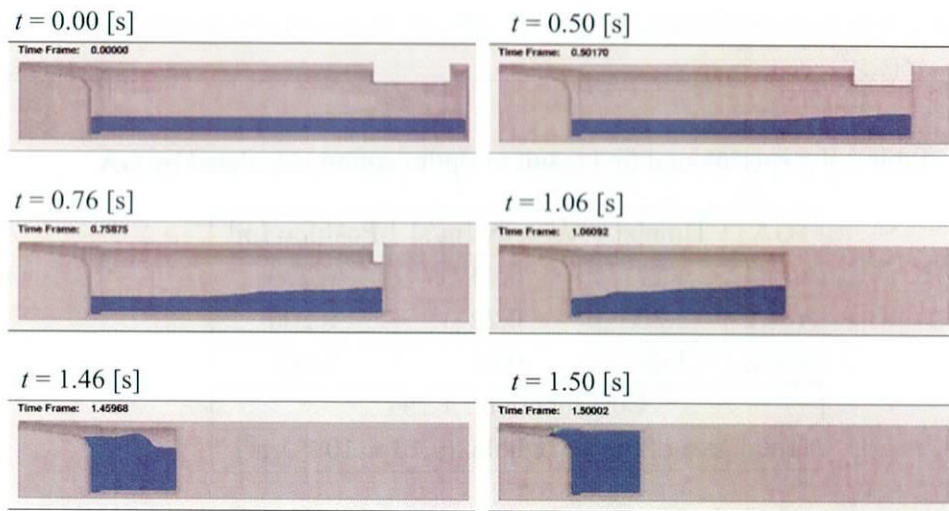


Fig. 5.12 Simulation result by using optimum velocity curve calculated by GA

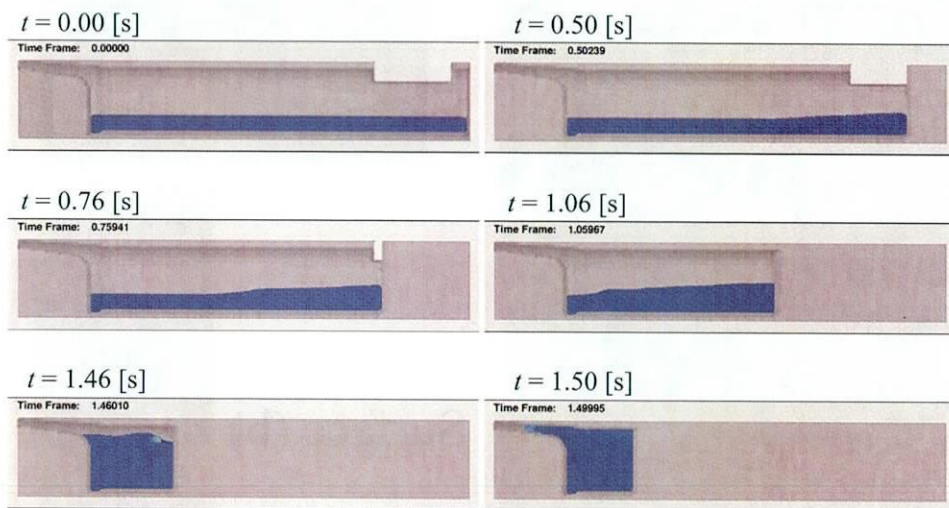


Fig. 5.13 Simulation result by using optimum velocity calculated by MSSA

5.6 実験結果

求まった最適プランジャ速度を実際のプラントに適用して、製品品質の評価を行った。最適化された速度パターンとその評価値および、実験における気泡面積を Table 5.4 と Table 5.5 に示し、ブリスト後の鋳肌表面の様子を Fig. 5.14 と Fig. 5.15 に示す。結果より、MSSA でも気泡面積が減少しており、遺伝的アルゴリズム同様に良好な実験結果を得られていることがわかる。

Table 5.4 Experimental first result of optimization calculated by GA

GA	Time[s]	Velocity [m/s]	Position [m]
$i = 1$	1.27	0.44	0.279
$i = 2$	1.30	0.22	0.290
$i = 3$	1.49	0.60	0.367
Cost function : 1.794			
Total area of the air bubble : $9.13 \times 10^{-2} \text{ [m}^2\text{]}$			

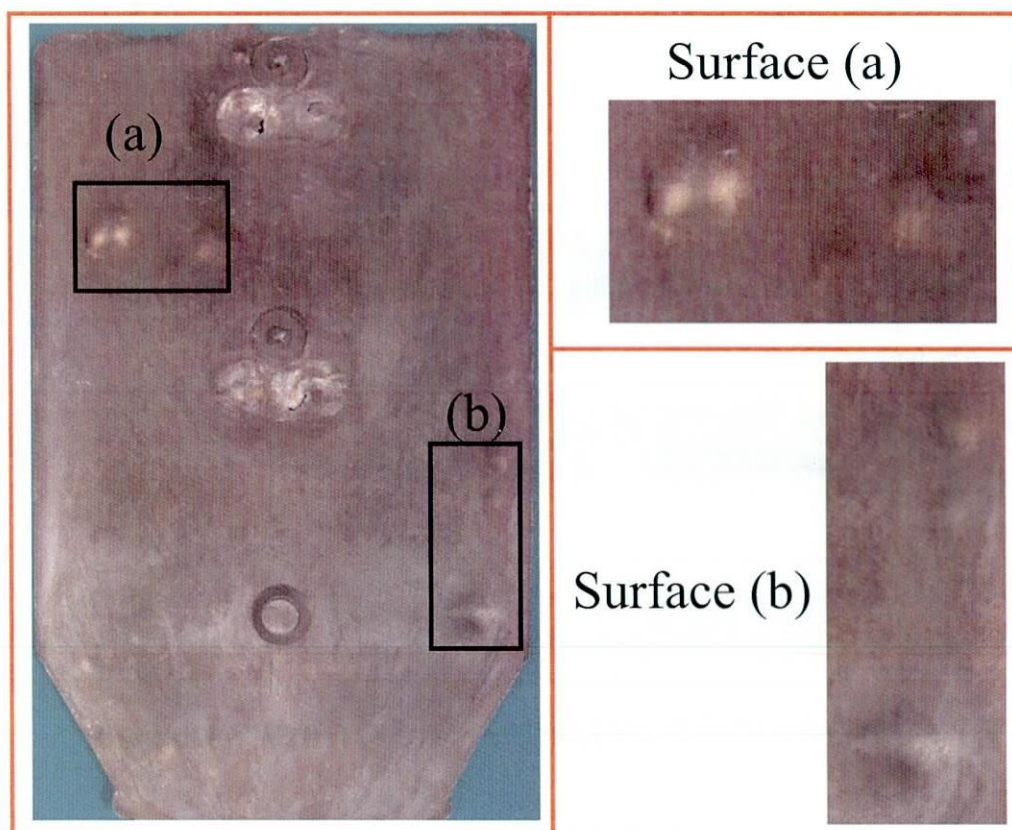


Fig. 5.14 Experimental result by using optimum velocity calculated by GA

Table 5.5 Experimental result by using optimum velocity calculated by MSSA

MSSA	Time [s]	Velocity [m/s]	Position [m]
$i = 1$	1.18	0.41	0.242
$i = 2$	1.44	0.34	0.340
$i = 3$	1.50	0.59	0.367
Cost function 1.789			
Total area of the air bubble : $7.79 \times 10^{-2} \text{ [m}^2\text{]}$			

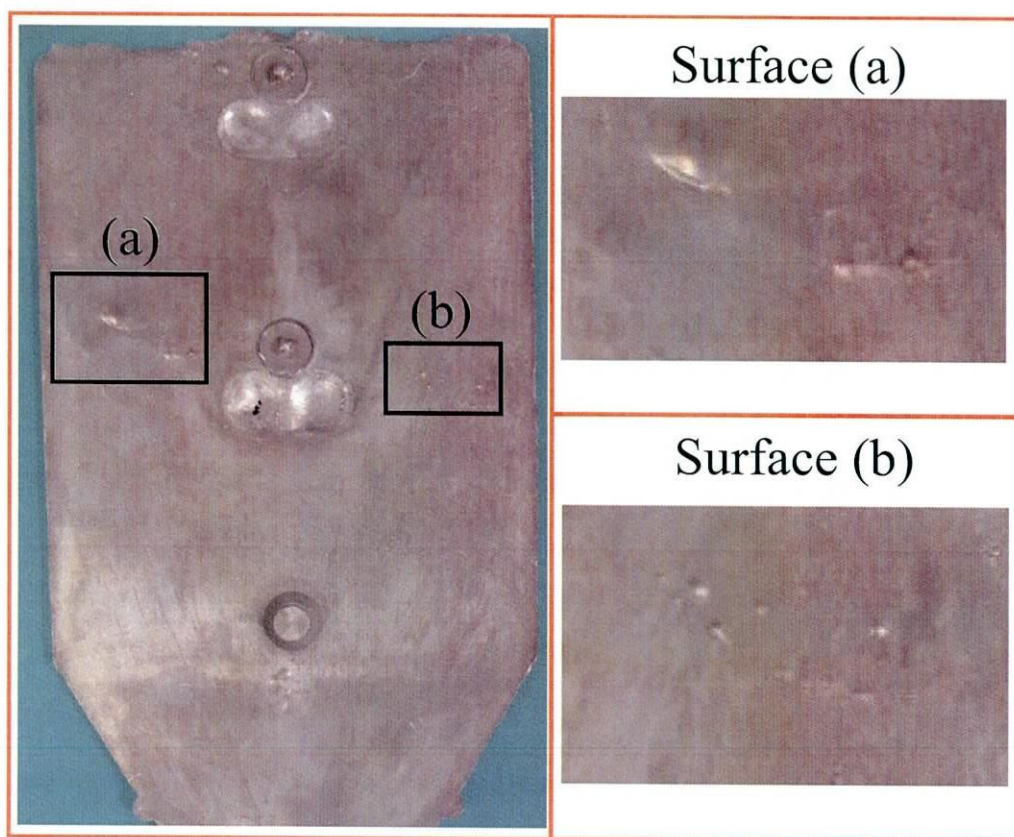


Fig. 5.15 Experimental result of optimization calculated by MSSA

5.7 まとめ

これまでに初期個体が収束性能に大きく影響を与えること、そして解の向上が見られる空間では集中的に探索点を配置させることで収束性能が向上することがわかった。そこで各探索点が他の位置や評価を考慮し、自ら良好な解空間へと向かう最適化アルゴリズムである多極集中型最適化アルゴリズムを構築した。結果、2変数のプランジャ最適速度において収束性能を比較したところ約50[%]の収束性能の向上が見られるとともに局所解への陥りも減少した。5変数のプランジャ最適速度にも同様に適用したところ、収束性能が向上したことが確認できた。また、実験を行ったところ、空気巻き込みの少ない良好な結果を得られたことから、MSSAの有効性を示すことができた。

第6章 クラスター分析を応用した空間探索アルゴリズムの構築

6.1 概要

MSSA では解探索に対する収束性能の向上を行うことができた。しかしながら算出された解は最小の評価値であり、EDSA のように大域的に良好な解を見つけているわけではない。そのため MSSA で算出された解を基に、大域的に良好な解を算出するアルゴリズムを適用することで、さらなる品質を向上させることが期待できる。

大域的に良好な解を見つけるために EDSA では近似曲線を用いた。これをもちいることで評価の傾向を予測して見つけることができた。しかしながら近似曲線を用いる場合、その次数を設定する必要があり、不用意な次数の設定は良好な結果を算出することが不可能となる。したがって大域的に良好な解を見つけるために近似曲線を用いるのは好ましくない。

Fig. 6.1 は 2 変数最適化問題に対して、MSSA が最適解到達までに配置した探索点のすべてを示しており、図中の破線で示している箇所は良好な解空間を示している。この図より、探索点が良好な解空間へと集中していることがわかる。特に、より大域的な空間には探索点の個体数が多いことがわかる。これは MSSA の集中アルゴリズムによるものであると考えられ、探索点の個体数を評価に取り入れることで近似曲線を用いずとも大域的に良好な解を算出することができると考えられる。本手法の有効性は、ダイカストにおけるプランジャ速度の最適化に適用して、実験結果を通して示すことにする。

6.2 クラスター分析

個体数の密集度を調べるに辺り、問題が多変数であることから、多変数の分布状況を知る手法である多変量解析を用いる。多変量解析とは個体同士が似通っている、もしくは似通っていないかを判断する解析手法の総称であり、様々な解析手法に分類される。Table 6.1 は多変量解析の分類分けを行った物である。なかでもクラスター分析や判別分析は標本の分類をすることができるため多変数であっても空間を分割することができる^{[72][73]}。

特に個体間の明確な空間分けができない場合を非階層構造と呼び、問題が非階層構造の場合、クラスター分析がよく使われる。非階層構造の場合、空間の大きさは個体密度によって変化できることが特徴である。したがって本章では多変量解析の一つであるクラスター分析の手法を応用し、大域的に良好な解空間を算出する。

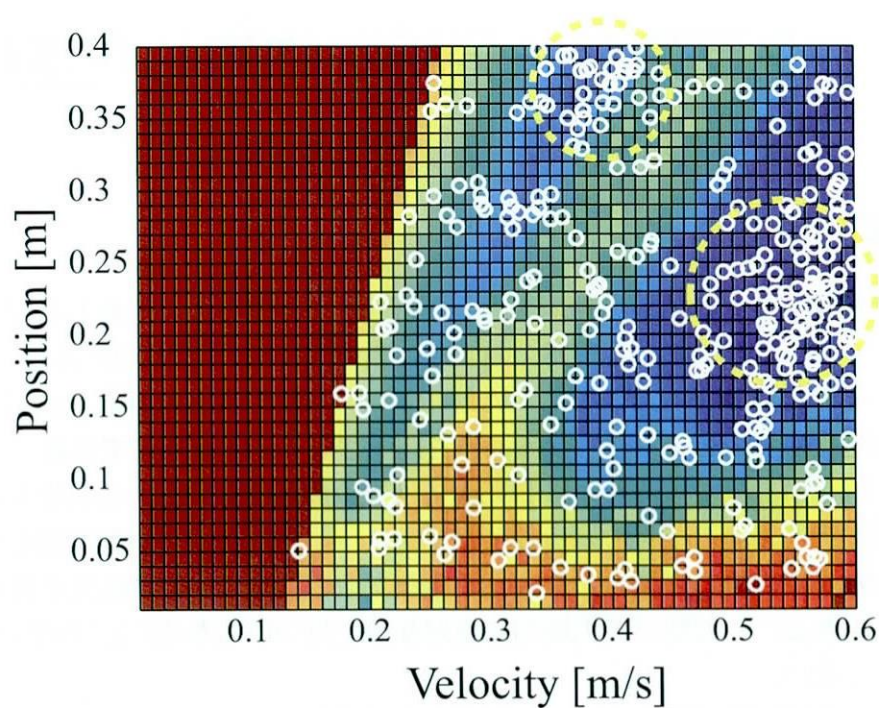


Fig. 6.1 Location of search points by using multi-subcenters solution search algorithm

Table 6.1 Classification of multivariate statistical methods

基準変数		多変量解析の目的	分析方法
あり	量的	予測式 (関係式の発見) 量の推定	重回帰分析 正準相関分析
	質的	標本の分類 質の推定	クラスター分析 判別分析
なし		多変量の統合整理 (減らす) 代表変量の発見 変数の分類	主成分分析 因子分析

6.3 空間探索アルゴリズムの設計

分析を行うにあたり、クラスター分析の概念を取り入れた大域的安定解算出法として空間探索アルゴリズム (Space Searching Algorithm : 以下 SSA と表記) を提案する。SSA は最適化完了後に適用され、分布状況の解析及び解の算出を行う。複数個の初期重心 g_{ini} を生成し、その付近にある探索点の数、距離と探索点群の平均の評価値を求める。その探索点数が形成するベクトルの重心 g_{shift} を求め、これを新たな重心として再度付近の探索点数の数などの情報を採取する。以後、重心の移動がなくなるまで、繰り返される。評価としては良好な解空間には探索点が集中することから、探索点数の数が優先され、次に平均の評価値が優先されるようにする。最終的に、探索点を最も多く含み、平均の評価値が低い解の重心を算出する。重心を求める式は Eq.(6.1) に示す。

$$g_c = \min \left\{ E \left(\sum_{\|g_c - p_h\| \in R_g} C_e(p_h) \right), \forall p_h \in S \right\} \quad (6.1)$$

ここで、 g_c は大域的に良好な空間の重心で、 E は平均的な評価値、 p_h は評価済みの探索点、 C_e はベクトル p_h の評価値、 R_g は許容誤差の範囲、そして S は評価済みの探索点群である。Fig. 6.2 は MSSA に SSA を適用した場合のアルゴリズムを表している。

6.4 探索結果

MSSA で得られた Table 5.5 の解に SSA を適用し、算出された速度入力を Table 6.2 に示す。評価値を比較すると適用後の方が悪く、充填速度は若干遅れ、また空気連行量も多いことがわかる。Fig. 6.4 は射出シミュレーションを示しており、シミュレーションから液体の挙動にも変化が見られないことがわかる。

Table 6.2 Simulation result using the MSSA result modified by SSA

Parameter	Value
Cost function J	2.158
Air entrainment A	0.329
Filling time	1.665 [s]

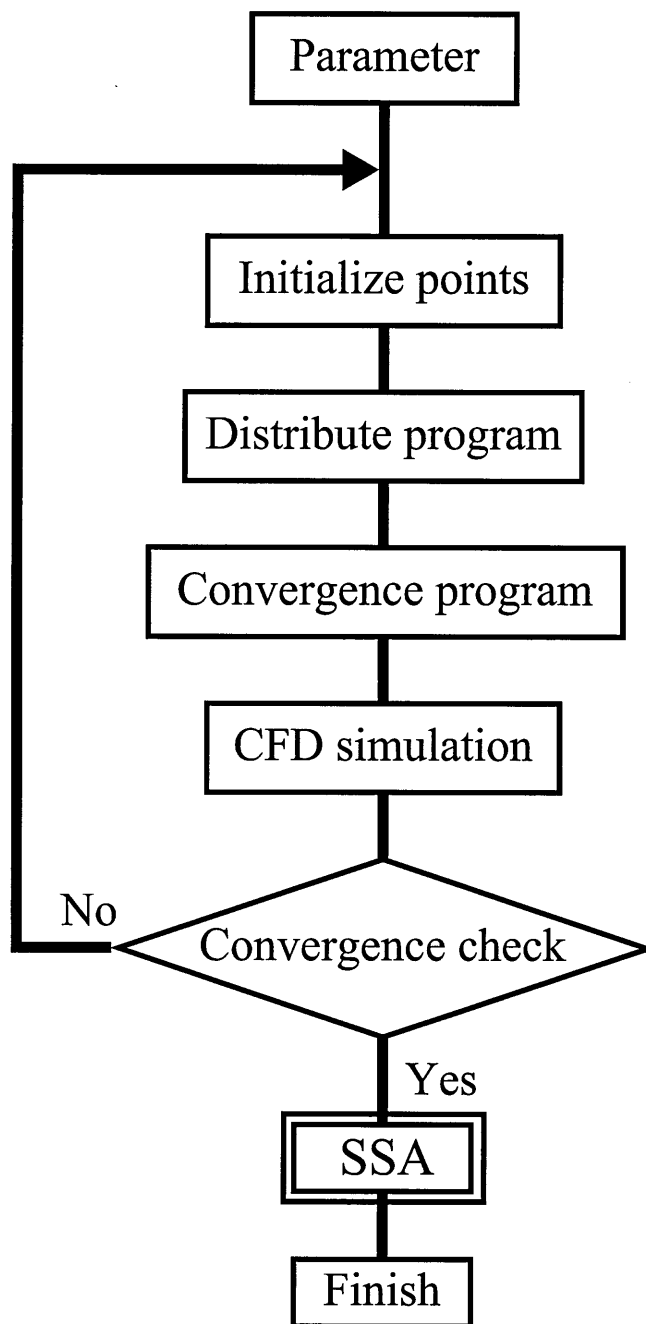


Fig. 6.2 Multi-subcenters Solution Search Algorithm adoption of Space Searching Algorithm

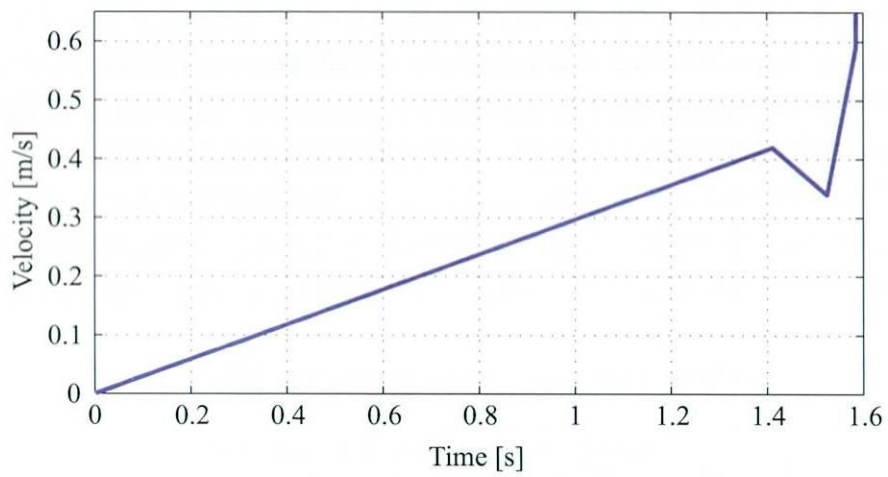


Fig. 6.3 Velocity curve by using the MSSA result modified by SSA

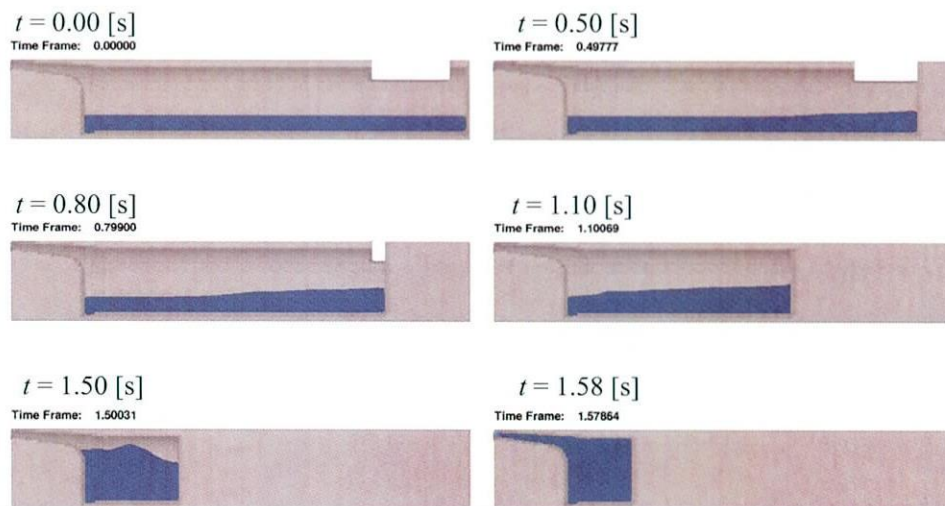


Fig. 6.4 Behavior of the fluid by using the MSSA result modified by SSA

6.5 実験結果

求まった最適プランジャ速度を用いて実験を行った。Table 6.3 は MSSA で導出された Table 5.5 で示された解に SSA を適用して導出した最適プランジャ速度となっている。比較すると速度変化はそれほど変化せず、しかしながら速度の切り替え位置が大きく変更していることがわかる。アルゴリズム適用前と比べて適用後はプランジャの射出完了時間が若干遅くなっていることがわかるが、評価関数においては適用したことで増加した。プリスタ試験の結果を Fig. 6.5 に示す。Table 5.5 と Table 6.3 から SSA を適用することで評価関数は増加しているが、その反面で気泡面積が適用前のものよりも気泡が少なくなっていることがわかる。これにより、SSA を適用することで品質の向上を狙える大域に安定な解を見つけることが期待できるとともに、SSA を適用することでシミュレーションの評価値に左右されず、良好な解を見つけることができる。

Table 6.3 Optimum velocity modified by SSA

	Time[s]	Velocity[m/s]	Position[m]
$i = 1$	1.41	0.42	0.296
$i = 2$	1.52	0.34	0.339
$i = 3$	1.58	0.59	0.367
Cost function 2.158			
Total area of the air bubble : $6.89 \times 10^{-2} \text{ [m}^2\text{]}$			

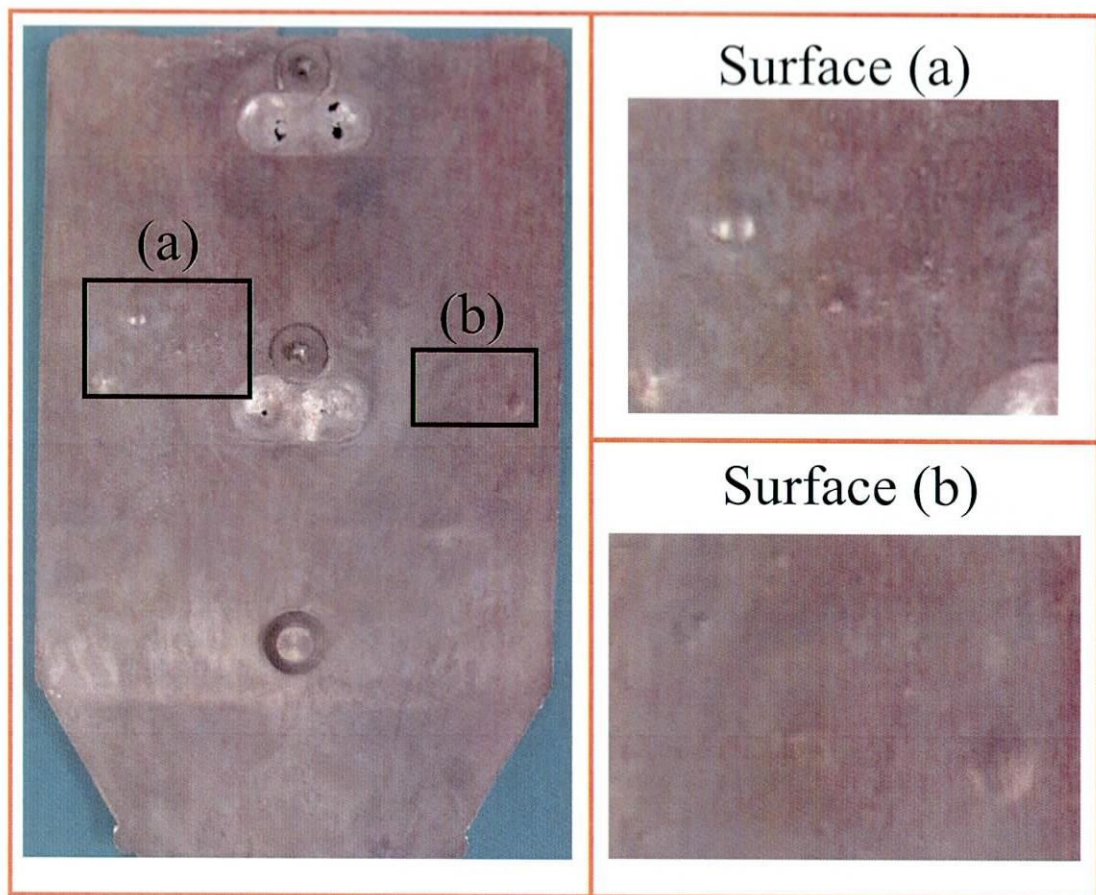


Fig. 6.5 Experimental result by using the MSSA result modified by SSA

6.6 まとめ

本章では大域的に良好な空間をとる解を算出することを目的として、良好な空間には多くの探索探索点が集まっていることから、クラスタ分析の理論を応用し、大域的に良好な評価値を取る解を見つける探索アルゴリズムの構築を行った。特に、MSSAは良好な解空間に探索点を集中することに長けているため、SSAを適用して解の算出を行った。適用前と適用後の最適化結果を実験結果を通して比較を行った結果、適用前よりも良好な実験結果を得ることができた。

第7章 結言と今後の展望と課題

7.1 結言

本研究ではCFDシミュレータを援用した最適化アルゴリズムの実用化を目的として、ダイカストにおける空気巻き込み低減を目指したプランジャ速度の最適化を中心とした、少ない計算回数かつ実験でも良好な結果を得られる解を算出することのできる最適化アルゴリズムの構築を行った。

CFDシミュレータを援用した最適化アルゴリズムとして、6自由度マニピュレータを用いた溢流抑制制御とダイカストにおける射出速度の最適化について述べた。6自由度マニピュレータを用いた溢流抑制制御では、CFDシミュレータを用いて液体の溢れを解析し、溢流や6自由度マニピュレータ等の制約をペナルティ項付きの最適化問題に定式化して、制約条件を満たしかつ高速な搬送を可能とする搬送速度の最適化を行った。ダイカストにおける射出速度の最適化では、CFDシミュレータを用いて欠陥原因の一つである空気巻き込み現象を解明、評価方法を構築し、高速かつ空気巻き込みを抑制する射出速度の最適化を行った。最適化アルゴリズムは遺伝的アルゴリズムを用いており、両者とも最適化により良好な結果を得ることができた。

しかしながらダイカストにおいては、最適化を行っても良好な結果を得られない場合があった。そこで大域的に数値が最小となるような解を算出するれば、実験でも良好な結果を得られるという考えに基づき、実験でも良好な結果を得られるような解の算出をすることの出来る最適化アルゴリズムの構築を目的として、極値近傍探索アルゴリズム(EDSA)を提案した。極値と近似曲線を利用して大域的に良好な評価を得ている空間を見つけ出すことで、実験でも良好な結果を得られる解の算出を行うことができた。また良好な解空間へ探索点を集中的に配置することで、結果的に収束性能が向上したことがわかった。

その一方で、近似曲線の次数の設定が収束性能や解の導出に影響を与えるため、他のプラントに適用など、多様性に欠ける。さらに、初期値が解の収束性能に大きく影響を及ぼすこと、そして解の向上が見られる空間に探索点を集中させることが収束性能の向上につながることから、分散制御則と集中アルゴリズムからなる多極集中型最適化アルゴリズム(MSSA)を構築した。これにより収束性能は大幅に向上するとともに、局所解への陥りも軽減することができ、良好な実験結果となった。

また実験でも良好な結果を得るために極値近傍探索アルゴリズム(EDSA)を提案し、近似曲線を用いて大域的に良好となる解の算出を行ったが、近似曲線を用いる場合、その次数を設定する必要がある、不用意な次数の設定は良好な結果を算出することが不可能となる。したがって大域的に良好な解を見つけるために近似曲線を用いるのは好

ましくない。その一方で、良好な解空間には多数の探索点が集中していることから、その密集度から大域的な最適解を導出する空間探索アルゴリズム (SSA) を提案した。多極集中型最適化アルゴリズム (MSSA) に適用した結果、良好な実験結果を得ることが可能となった。

最終的に、多極集中型最適化アルゴリズム (MSSA) と空間探索アルゴリズム (SSA) を複合させた最適化アルゴリズムを用いることで解を効率良く求めることができ、さらに実験でも良好な評価を得ることができる (検証については付録Fを参照)。また、今回適用したダイカストだけではなく、重力鋳造における傾動速度の最適化や船舶の形状最適化など様々な設計最適化支援ツールとして用いられることが期待でき、生産準備段階におけるコストの削減や、設計期間の短縮などの効果が期待できる。

7.2 今後の課題と展望

今後の課題と展望として、最適化における最適解算出する時間をさらに短縮することにある。提案した手法では遺伝的アルゴリズムだと54世代まで掛かるものを39世代まで抑えることができた。しかしながら、1世代に用いた探索点数は30個であり、39世代まで掛かる物だと仮定すると $30 \times 39 = 1170$ 回の計算が必要となってくる。今回対象としたプラントは比較的単純であり、1回に掛かる計算時間が1～5分であったため有限時間内に最適化を終えることができた。しかしながら対象とするプラントが複雑である場合、例えば重力鋳造などでは1回の解析に1時間程度を要し、このようなプラントを用いた場合、解の算出までに非常に長い時間を要してしまう。したがって5世代程度 ($30 \times 5 = 150$) で終えるような最適化アルゴリズムを構築することが課題となってくる。今回提案した手法は主に配置の問題を中心にアルゴリズムを構築して最適化を行ったが、CFDシミュレータそのものの特徴、例えばメッシュの設定や、VOF法やFAVOR法における計算誤差、体積損失、エネルギー損失など、多峰性解空間を生む要素そのものを考慮に入れ、良好な解空間の予測、配置の設定を行うことで収束性能を大幅に向上させることが考えられる。つまり、1つの探索点で得られた解析データを基に探索点を適切に配置させることのできる最適化アルゴリズムの構築を行う必要がある。

また本論文で最適化の対象としたのがダイカストにおけるプランジャの最適速度の導出や重力鋳造における傾動入力値の最適化、そして溢流抑制軌道の生成であったが、CFDシミュレータを援用した最適化の要望は多く、例えば、ダイカストにおける金型やプランジャチップの形状の最適化は、ただ単に今回のような速度や空気巻き込み量を評価するだけではなく、強度や加工条件などを加える必要があり、非常に複雑な問題となる。特に今使用されているプランジャチップの形状は平形と凸形状のたった2種類であるため、その最適形状の導出が非常に望まれている。

鋳造分野以外にも洗浄能力や節水性能を向上させた便器形状の最適化や、ダムにおけるゲート部の最適化、水力発電におけるタービンブレードの最適形状など様々な分野への応用が期待できる。

学術論文と国際会議

学術論文

1. K.Yano, K.Hiramitsu, Y.Kuriyama and S.Nishido, "Optimum Velocity Control of Die Casting Plunger Accounting for Air Entrapment and Shutting", International Journal of Automation Technology, Vol. 2, No. 4, pp.259-265, 2008
2. 栗山嘉文, 矢野賢一, "溢流の防止を考慮した液体ハンドリング制御", 日本機械学会論文集C編, 第75巻, 第754号, 1690頁～1697頁, 2009年
3. 栗山嘉文, 前田将宏, 矢野賢一, 道岡裕也, "アルミニウム合金の重力金型鑄造における多段階注湯制御入力最適化", 鑄造工学, 第82巻, 第9号, 531頁～537頁, 2010年
4. Yoshifumi Kuriyama, Ken'ichi Yano, "Multi-subcenters Solution Search Algorithm for CFD Optimization Problems and Its Application to Die Casting", Material Transactions, (投稿中)
5. 栗山嘉文, 林脩一郎, 矢野賢一, 渡邊守, 西戸誠志, "CFD最適化問題のための極値近傍探索アルゴリズムの構築とダイカストプロセスへの適用", 鑄造工学, (投稿中)

査読つき国際会議論文

1. Y. Kuriyama, Y.Yasuda, K.Yano and M. Hamaguchi, "Liquid Container Transfer Control by Hybrid Shape Approach for 6DOF manipulator", Proc. of SICE Annual Conference, pp. 3069-3072, Takamatsu, Japan, 17-20 September, 2007
2. Y.Kuriyama, K.Yano and M. Hamaguchi, "Trajectory Planning for Meal Assist Robot Considering Spilling Avoidance", Proc. of IEEE MSC(IEEE Conference on Control Applications), pp. 1220-1225, San Antonio, USA, 3-5 September, 2008
3. K.Yano and Y.Kuriyama, "3-D Transferred Control using Spilling Avoidance Control for Meal Assist Robot by using CFD Simulator", Proc. of The 10th International Conference on Control, Automation, Robotics and Vision (ICARCV 2008), pp. 1758-1763, Hanoi, Vietnam, 17-20 December, 2008

4. Y.Kuriyama, K.Yano and S.Nishido, "Optimum injection considering shot timing for Die casting by using CFD simulator", Proc. of SICE Annual Conference, pp. 815-819, Hakata, Japan, 18-21 August, 2009
5. Y.Kuriyama, S.Hayashi, K.Yano and M.Watanabe, "Solution search algorithm for a CFD optimization problem with multimodal solution space", Proc. of IEEE CDC (Conference on Decision and Control), pp. 5556-5561, Shanghai, China, 16 - 18 December, 2009
6. Y.Kuriyama, K.Yano and Y.Suzuki "Development of solution search algorithm for a CFD optimization problem", Proc. of KSEEE-JSAEM 2010 International Engineering conference, pp. 89-93, Nairobi, Kenya, 5 - 6 August, 2010
7. Y.Kuriyama, K.Yano S.Nishido and M.Watanabe, "Optimization of die-casting plunger velocity using distribute control algorithm", 69th World foundry congress, pp.550-553, Hangzhou, China, 16 - 20 October, 2010
8. K.Yano, M.Maeda,Y.Kuriyama and S.Nishido, "Optimization of Multistep Pouring Control Input for Aluminum Gravity Casting", 69th World foundry congress, pp.229-232, Hangzhou, China, 16 - 20 October, 2010
9. Y.Kuriyama and K.Yano, "Optimal algorithm to multimodal solution space formed by CFD simulator", 18th IFAC World Congress, Milano, Italy , August 28 - September 2, 2011, (投稿中)

国内学会における発表

1. 栗山嘉文, 矢野賢一, 6自由度マニピュレータによる液体搬送制御, 計測自動制御学会中部支部統合知能メカトロシステム講演会, 岐阜, 2006年11月3日
2. 栗山嘉文, 矢野賢一, 浜口雅史, 6自由度マニピュレータによる液体搬送制御系設計, 第7回計測自動制御学会システムインテグレーション部門学術講演会, 札幌, 2006年12月14-17日
3. 栗山嘉文, 矢野賢一, 西戸誠志, ダイカストの品質向上を目的とした多段階射出速度制御, 第153回日本鑄造工学会全国講演大会講演概要集, 金沢, 2008年10月24-27日
4. 矢野賢一, 栗山嘉文, 食事支援ロボットによる液体ハンドリング制御, 第51回自動制御連合講演会, 山形, 2008年11月22-23日
5. 栗山嘉文, 矢野賢一, ダイカストの品質向上を目的とした多段階射出速度制御, JFS東海YFEフォーラム'08, 名古屋, 2008年11月28日
6. 林脩一郎, 栗山嘉文, 矢野賢一, 渡邊守, 西戸誠志, 多峰性解空間を持つCFD最適化問題のための解探索アルゴリズムの構築とダイカストプロセスへの適用, 第154回日本鑄造工学会全国講演大会講演概要集, 東京, 2009年5月29日-6月1日

7. Y.Kuriyama, K.Yano, Y.Nemoto and S.Nishido, Optimization of Pouring Control Input using Decentralized Control Algorithm, 第156回日本鑄造工学会全国講演大会(日韓 YFE 大会)講演概要集, 長崎, 2010年5月21-24日
8. 栗山嘉文, 矢野賢一, 渡邊守, 多峰性解空間に対する最適化アルゴリズムの構築, 日本鑄造工学会東海支部 YFE フォーラム'09, 名古屋, 2010年1月22日
9. 栗山嘉文, 矢野賢一, 月岡佑介, 西戸誠志, アルミニウム重力金型鑄造における最適傾動速度制御, 日本鑄造工学会東海支部 YFE フォーラム'09, 名古屋, 2010年1月22日
10. 栗山嘉文, 矢野賢一, 根本泰則, 西戸誠志, 分散制御則を利用したCFD最適化アルゴリズムの構築とダイカストプランジャ速度制御の応用, 第156回日本鑄造工学会全国講演大会講演概要集, 大阪, 2010年5月21-24日
11. 金澤賢一, 栗山嘉文, 矢野賢一, 根本泰則, 西戸誠志, CFD解析に基づくダイカストスプルー形状の最適化, 第157回日本鑄造工学会全国講演大会講演概要集, 札幌, 2010年10月1-4日

特許

1. 発明者: 矢野賢一, 栗山嘉文, 出願人: 岐阜大学, "液体搬送装置", 特願2008-224130, 2008年9月1日

著書

- K.Yano, Y.Kuriyama, "Extremal Distribution Sorting Algorithm for a CFD optimization problem", InTech -Search Algorithms-(掲載決定)

その他講演

1. 栗山嘉文, 食事支援ロボットによる液体ハンドリング制御, Flow-3D Conference 2008, 東京国際フォーラム, 2008年10月31日
2. 栗山嘉文, 多峰性解空間を持つCFD最適化問題のための解探索アルゴリズムの構築とダイカストプロセスへの適用, Flow-3D Conference 2009, 東京国際フォーラム, 2009年10月30日
3. Y.Kuriyama, "Optimum injection control for Die casting by using CFD simulator", Proc. of 6th Joint Symposium between Faculty of Engineering, Gifu University and College of Engineering, Chonnam National University, Korea, 26-29 November, 2009
4. 栗山嘉文, 流体挙動シミュレータを援用した最適化アルゴリズムの開発, 第156回日本鑄造工学会全国講演大会講演概要集 新東工業奨学生講演, 大阪, 2010年5月21-24日

5. 栗山嘉文, 矢野賢一, CFDを援用した最適化に対する収束性能向上を目的とした探索アルゴリズム, 日本鑄造工学会東海支部 YFE フォーラム'10, 名古屋, 2010年11月19日

謝 辞

本論文は、平成20年4月から平成23年1月末までに遂行された研究をまとめた物であり、多くの先生方のご指導と多くの方々のご協力のたまものであります。ここに感謝の意を表します。

研究室配属当初から指導教官であった三重大学大学院工学研究科機械工学専攻教授(旧：岐阜大学工学部人間情報システム工学科准教授)の矢野賢一先生には、大変お世話とご迷惑をお掛けしましたが、それでも大変お忙しい中、ご指導をいただきました。また、やりがいのあるテーマを頂いただけでなく、本論文ならびに他の投稿論文に関するご指導、特に輪講やミーティングに関しては深く追求すること、客観的に自分の研究を見直すことの必要性、研究生一人一人への思いやりと厳しさ、また、研究者としての心構えや社会貢献への意義など研究者としての考え方についてもいろいろとご指導頂きました。大志を抱き、その夢を実現させるためにコツコツと研究することを、自身の研究姿勢で奨励したくださったことに深く感謝します。そして数回にわたる国内講演会と6度の国際会議への参加、5週間にわたる海外への出張、企業とのプロジェクト会議など多くの社交や議論の場を与えて頂き、見聞を広められたことに厚くお礼申し上げます。本論文を遂行だけでなく、これから岐阜工業高等専門学校助教として、社会貢献できる自身と指導する自身がついたことも、偏に矢野先生のご指導の賜であり、深く感謝致します。また三重大学の教授になられたことを心よりお祝い申し上げます。

ご多忙にもかかわらず、本論文の審査委員を務めて頂き、有益なご助言と指導を頂きました本学工学部人間情報システム工学科教授 山田宏尚先生、教授 野方文雄先生、教授 佐々木実先生に厚くお礼を申し上げます。

FLOW-3Dのソフトウェアに関して、適切なアドバイスをしていただいた、(株)テラバイトの根本様、小賀様、渡辺様、また実験作業において、アイシン高丘(株)の故・初山様、西戸様、西川様、道岡様には多忙な中で研究協力者として携わっていただき、ここに感謝の意を表したいと思います。

出張や物品購入に辺り様々なサポートをして頂きました仁科様には大変お世話になり大変感謝しております。

インターンシップの機会を与えて頂き、ご指導して頂きました水谷様を始め、岐阜県機械材料研究所の皆様には大変感謝しております。

研究の遂行にあたりご協力していただいた卒業生、平光氏、林氏、前田氏、吉村氏に感謝の意を述べます。またミーティングにおいて様々な助言を頂きました、ジャニス工業(株) 鈴木様、月岡氏、高木氏、堀江氏、金澤氏に感謝の意を述べます。研究会

等においてご指導，ご助言を頂いた卒業生・菰口先輩を始め，研究室の皆様に心からお礼申し上げます。

また私を数学の道，工学の道に導いてくださいました学研 CAI スクールの前田先生に心よりお祝い申し上げます。

最後に，博士課程が修了する二七歳までの間，辛き時を支え，勉学の機会を与えて頂き，有意義な学生生活を送らせて頂くとともに，様々な経験をさせて頂きました両親，祖父，祖母，弟，妻，親戚一同に深く敬意を表します。

皆様の多大なるご支援，ご鞭撻の下に今日まで遂行してこれたことを大変感謝すると共に，今後は私が培った経験，教育法をもとに，社会や多くの方々に貢献していく所存です。

参考文献

- [1] 高橋伸吾, 山田実, “2次元システム制御による繰り返し注湯流量制御” 鑄造工学会 第155回全国講演大会, (2009), pp.80
- [2] 福嶋隆輔, 野田善之, 寺嶋一彦, 鈴木新雄, 牧野泰育, “自動注湯機における注湯推量推定に基づく取鍋流出液体落下位置制御” 鑄造工学会 第155回全国講演大会, (2009), pp.136
- [3] 矢野賢一, 杉本 祐, 野田善之, 寺嶋一彦, “Betterment Process による円筒取鍋型自動注湯ロボットの注湯流量制御” 日本機械学会論文集 (C編), 70巻, 694号, 2004, pp.206–pp.213
- [4] 矢野賢一, 金子元樹, 野田善之, 寺嶋一彦, 川合悦蔵, “スラグ付着の影響を考慮した自動注湯ロボットのインテリジェント制御” 鑄造工学会, 78巻, 第8号, (2006), pp.383–pp.388
- [5] 矢野賢一, 松尾祐介, 野田善之, 寺嶋一彦, 川合悦蔵, “扇形取鍋を用いた自動注湯プロセスのモデリングと注湯制御入力導出法” 鑄造工学会, 78巻, 第6号, (2006), pp.297–pp.303
- [6] 野田善之, 寺嶋一彦, “容器傾動により流出する液体の流量モデルと制御” Proc. of the 50th Annu. Conf. of the ISCIE, Kyoto, May 10-12, 2006, pp.101–pp.102
- [7] 野田善之, 寺嶋一彦, “実用取鍋を用いた自動注湯プロセスのモデリングと注湯流量制御”, Proc. of the 2006 JSME conf. on Robotics and Mechatronics, Waseda, Japan, May, 26-28, 2006, 1P1-D35(1)–1P1-D35(4)
- [8] 矢野賢一, 金子元樹, 杉本祐, 寺嶋一彦, “注湯プロセスにおける熟練技能を実現する自動注湯ロボットのスーパーバイザリ制御” 日本ロボット学会誌, Vol.21, No.6, (2003), pp.670–pp.681
- [9] 寺嶋一彦, 矢野賢一, 金子元樹, 三好孝典: 注湯プロセスにおける熟練技能を実現する自動注湯ロボットのスーパーバイザリ制御: 豊橋技術科学大学卒業論文 (2003)
- [10] 矢野賢一, 戸田貴弘, 東川晋平, 寺嶋一彦, “スロッシング抑制を考慮した自動注湯ロボットの制御” 日本機械学会論文集 (C編), 69巻, 678号, (2003), pp.387–pp.395
- [11] 寺嶋一彦, 矢野賢一, 杉本祐, 渡辺充昭 “傾動式自動注湯機における湯口カップ内液面レベル制御” 鑄造工学 Vol. 73, No. 5(2002)

- [12] 矢野賢一, 杉本祐, 寺嶋一彦, 渡辺充昭 “傾動式自動注湯機における湯口カップ内液面レベル制御” 鑄造工学 Vol. 74, No. 3(2002), pp.163–pp.169
- [13] Kazuhiko Terashima, Ken'ichi Yano, Yu Sugimoto and Mitsuaki Watanabe, "Position control of ladle tip and sloshing suppression during tilting motion in automatic pouring machine" 10th IPAC symposium on automation in mining, MMM2001, sep, 2001, tokyo, pp.182–pp.187
- [14] Kazuhiko Terashima, Ken'ichi Yano, "Sloshing analysis and suppression control of tilting-type automatic pouring machine" Control engineering practice 9, (2001), pp.607–620
- [15] 矢野賢一, 寺嶋一彦, 稲垣知親, “2自由度制御による取鍋内スロッシング制振制御” 鑄造工学会論文誌, 第73巻, 第5号, (2001), pp.299–pp.304
- [16] 寺嶋一彦, 矢野賢一, 浜口雅史, 藤原邦治, 鈴木祐二, “流動, 振動プロセスの最適化と制御” 鑄造工学会学会誌, 第72巻, 第3号, (2000), pp.204–pp.216
- [17] 能勢和夫, 小西正躬, “システム制御理論の連鑄湯面レベル制御への応用” 鑄造工学会学会誌, 第72巻, 第4号, (2000), pp.281–pp.287
- [18] 兼重明宏, 荻智成, 榎本佳祐, 兼重直樹, 三好孝典, 寺嶋一彦, “天井クレーンによる液体タンク搬送制御系設計に関する研究(タンク内液体挙動の解明とロッド・タンク連成系の最適設計の検討)” 鑄造工学会 第155回全国講演大会, (2009), pp.79
- [19] 渋谷涼太, 野田善之, 寺嶋一彦, “液体容器形状とその傾動状態に対する液面振動解析と制振制御” 鑄造工学会 第155回全国講演大会, (2009), pp.137
- [20] Kazuhiko Terashima and Tomochika Inagaki, "Feedforward and Feedback Control on Liquid Vibration and Rotary Angle in the Tilting Motion of Automatic Pouring" Proc. of the 2nd Asian Control Conf., July, (1997), Seoul, pp.375–pp.378
- [21] Kazuhiko Terashima, Masafumi Hamaguchi and Kazuto Yamaura, "Modeling and Input Shaping Control of Liquid Vibration for an Automatic Pouring System" Proc. of the 35th CDC, (1996), Kobe, Dec, pp.4844–pp.4850
- [22] Kazuhiko Terashima, Akihiro Kaneshige, Hiroshi Koyamatsu, "Modeling and Motion Control of Fluid Behavior in Automatic Pouring System" IFAC 13th Triennial World Congress, San Francisco, pp.403–pp.408
- [23] 寺嶋一彦, 浜口雅史, 兼重明宏, “鑄物工場自動化のための制御設計技術と応用” 鑄造工学会学会誌, 第69回, 第3号, (1997), pp.257–pp.265

- [24] 加納学, 長谷部信治, 橋本伊織, 大野弘, “2 型モデル予測制御系の設計” 第 6 回制御技術シンポジウム, (1998), pp.71–pp.76
- [25] 勝部展生, 永井正夫, “傾動式自動注湯機の液面制御に関する研究 (第 1 報, モデリングと制御)” 日本機械学会論文集 (C 編), 第 65 巻, 634 号, (1999), pp.2345–pp.2351
- [26] 矢野賢一, 寺嶋一彦: 静止液位変動に対する公称モデルの検討と液体ロバスト搬送制御: システム制御情報学会論文誌 C 編, Vol. 10, No. 8(1997)
- [27] 矢野賢一, 寺嶋一彦: LMI に基づく極配置制約付き H_2/H_∞ 制御による液体搬送制御: 日本機械学会論文集 (C 編), Vol. 64, No. 625(1998)
- [28] 寺嶋一彦, 矢野賢一, 小黒直樹: 時間, 周波数仕様を考慮したハイブリッド整形法による制振起動制御: 計測自動制御学会論文集, Vol. 35, No. 5(2001)
- [29] 大林茂, CFD 利用の新段階: 数値最適化 (小特集: CFD と最適設計), 日本機械学会誌, vol.105, no.999, (2002), pp.64–69
- [30] 満田正彦, 細川佳之, 織田剛, 小林利行, 赤松博史, 山田斉, 数値流体力学 (CFD) が設計者に開く新しい世界, 神戸製鋼技報, vol.51, no.3, (2001), pp.69–73
- [31] 大中逸雄: “シミュレーションの最近の動向”, 鑄造工学, VOL. 78, NO. 12, pp. 602–608, 2006
- [32] 天野研, 岩野龍一郎, “噴流を伴う容器内のスロッシング現象の実験と解析”, 日本機械学会論文集 (B 編), (1991), 第 57 巻, 538 号, pp.21–pp.28
- [33] 田中宏明, 中山司, “3 次元貯槽内の非線形スロッシングに対する境界要素解法”, 日本機械学会論文集 (B 編), (1991), 第 57 巻, 538 号, pp.8–pp.13
- [34] K.Terashima, K.Yano, H.Kawagishi, K.Okamura, and M.Watanabe, “An optimum design with sloshing suppression on container transfer control of molten metal using computer fluid dynamics model”, International Journal of Cast Metals Research, Vol. 15, No. 4, pp. 435–440, 2002
- [35] 大中逸雄, 杉山明, 湯流れの直接観察と数値シミュレーション, “日本鑄造工学会第 151 回全国講演大会講演概要集”, (2007), pp.80
- [36] 遠藤健介, 鈴木克美, 鈴木雅也, 櫻井大八郎, 木島秀弥, ダイカストスリーブ内溶湯挙動の可視化とシミュレーションのための予備実験, “日本鑄造工学会第 151 回全国講演大会講演概要集”, (2007), pp.79
- [37] 久保公雄, 佃公博, 朝尾浩光, 出来尚隆, 精密鑄造法によるタービンブレードの湯流れ・凝固・組織予測シミュレーション, “日本鑄造工学会第 151 回全国講演大会講演概要集”, (2007), pp.108

- [38] 出来尚隆, 佃公博, 久保公雄, 金型用水冷管の温度及び熱伝達係数の分布を考慮したダイカスト鑄造シミュレーション, “日本鑄造工学会第 151 回全国講演大会講演概要集”, (2007), pp.107
- [39] 野口徹, 堀川紀孝, 佐々木真, 中村孝, 鑄造シミュレーションによる鑄ぐるみ接合の方案検討とその実験的検証, “鑄造工学”, Vol.78, No.12, (2006), pp.668-674
- [40] 別府雅弘, 金沢憲一, 村越茂, 遠藤宏光, ダイカストにおける湯流れ凝固シミュレーションの利用技術, “日本ダイカスト会議論文集”, (2002), pp.125-130
- [41] 久保公雄, ダイカストの湯流れ及び凝固シミュレーション, “鑄造工学”, Vol.78, No.12, (2006), pp.620-628
- [42] 李紹敏, 峯憲一郎, 三中西信治, 安斎浩一, ダイカスト離型剤スプレーモデルによる金型冷却の数値シミュレーション, “鑄造工学”, Vol.79, No.10, (2007), pp.571-578
- [43] Mi Guofa, Liu Xiangyu, Wang Luangfei, Fu Hengzhi, “Numerical simulation of low pressure die-casting aluminum wheel” China foundry, vol.6, no.1, pp.48–pp.52 february, 2009,
- [44] 杉山明, 大中逸雄, 岩根潤, 安田秀幸, 砂型での溶湯充てん過程の直接観察と数値シミュレーション, “鑄造工学”, Vol.78, No.12, (2006), pp.691-697
- [45] 恩田祐, 西川真一, 村上弘文, 神戸洋史, 大中逸雄低温鑄込み ADC14 合金ダイカストの湯流れ模様とポロシティ, “鑄造工学”, Vol.75, No.1, (2003), pp.17-22
- [46] 坂村正行, 菊池政男, 高橋勇, 岡本賢一, 村上工成, 田中元基 “湯流れ挙動に及ぼすランナー・ゲート形状の影響”, JDCA, Vol2, No.19, pp.113–pp.124
- [47] 飯見秀紀, 亀山吉朗, 鈴木芳雄, 長縄由紀子, 加藤鋭次, 野村宏之ダイカスト湯流れの可視化検証による解析精度向上, “デンソーテクニカルレビュー”, Vol.6, No.2, (2001), pp.100-106
- [48] 三中西信治, 藤田和美, 射出スリーブ内湯流れシミュレーション, “日本ダイカスト会議論文集”, (2000), pp.179-183
- [49] 木間塚明彦, 黒木康徳; “鑄物のガス欠陥予測シミュレーション”, 石川島播磨技報, VOL. 47, NO. 1, pp. 25–31, 2007
- [50] 柳浦睦憲, 茨木俊秀, 組合せ最適化 -メタ戦略を中心として-, 朝倉書店, 第一刷, (2001)
- [51] 貝原俊也, 藤井信忠, 指尾健太郎, 長谷川博之, 黒瀬伸二, 組み合わせオークションを用いた最適化手法に関する研究-リエントライトフローショップにおける資源割当問題への適用-, ISICE, Kyoto, (2008), pp.497-498

- [52] 喜多一, 遺伝的アルゴリズムによる最適化の現状, 若手研究者・学生向けに最新技術をわかりやすく紹介する講演会「確率的アルゴリズムによる情報処理」, 講義ノート, <http://www.smapip.is.tohoku.ac.jp/smapip/2003/tutorial/textbook/hajime-kita.pdf>, (2003)
- [53] 平田信行, "高速船の形状最適化", 第23回数値流体力学シンポジウム, Vol.23, pp.C8-2, 2009
- [54] 川面恵司, 横山正明, 長谷川浩志; "最適化理論の基礎-GA および MDO を中心として-", コロナ社, 2000
- [55] 章宏, 石川眞澄, 局所探索を用いたハイブリッド遺伝的アルゴリズムの試み, 電子情報通信学会技術研究報告, vol.104, no.474, (2004), pp.59-64
- [56] 片山謙吾, 成久洋之, 大規模な最大多様性問題に対する遺伝的局所探索, 情報処理学会論文誌: 数理モデル化と応用, vol.45, no.SIG 2(TOM 10), (2004), pp. 99-109
- [57] 筒井茂義, アントコロニー最適化手法, 計測と制御, 第47巻, 第6号, (2008), pp.466-472
- [58] 片山謙吾, 石渕久生, Memetic アルゴリズム, 計測と制御, 第47巻, 第6号, (2008), pp.487-492
- [59] 小笠原 英穂, 矢部博, "制約付最適化問題に対する逐次二次計画法における更新行列のサイジング", 数値解析研究所講究録, 915 巻 1995, pp.203-208
- [60] Donald R.Jones, Matthias Schonlau, and William J.Welch, "Efficient Global Optimization of Expensive Black-Box Functions" Journal of Global Optimization Vol.13:455-492,1998
- [61] 木下峻一, 佐久間淳, 小林重信, 小野功, UV 構造を考慮した実数値 GA の新たな枠組みの提案とその性能評価, 2008 年度人工知能学会全国大会 (第 22 回) 論文集, (2008), 2B1-4
- [62] 金光秀雄, 新保勝, 二変数多峰性関数の最小値探索アルゴリズム, 情報処理学会研究報告.MPS, 数理モデル化と問題解決研究報告, vol.95, no.44, (1995), pp.41-48
- [63] 正道寺勉, 多変数多峰性関数に対する最適値探索法の研究, 日本オペレーションズ・リサーチ学会論文誌, vol.20, no.4, (1977), pp.311-320
- [64] Harvey M Thompson, Carl A Gilkeson, Vassili.V.Toropov, Philip H.Gaskell, Mark C.T.Wilson, "CFD Optimization of Small Livestock Trailers " V European Conference on Computational Fluid Dynamics (ECCOMAS CFD 2010), June 2010

- [65] 矢野賢一, 前田将宏; “空気取り込み量を考慮した高速揺動攪拌制御”, 化学工学論文集, 第 34 巻, 第 4 号, pp. 424-431, 2008
- [66] 三宮信夫, 喜多一, 玉置久, 遺伝的アルゴリズムと最適化, 岩本貴司, 朝倉書店, 第一刷 (1998)
- [67] 桜木卓也, 中山一陽, ダイカストにおける湯境低減のための射出速度の最適化, “*鑄造工学*”, Vol.79, No.10, (2007), pp.592-599
- [68] 斉藤侑利子, 木間家明彦, 黒木康德, 凝固シミュレーションとタグチメソッドによる鑄造条件の最適化, “日本鑄造工学会第 151 回全国講演大会講演概要集”, (2007), pp.81
- [69] iSIGHT, Engineous Software, Inc. <http://www.engineous.jp/isight/>
- [70] 高橋勇, 安斎浩一, 森春幸, 佐々木幹夫, ダイカストプロセスにおける自動最適化シミュレーション手法の適用例, “日本ダイカスト会議論文集”, (2002), pp.107-112
- [71] K.Yoshimura, K.Yano, T.Fukui, S.Yamamoto, S.Nishido, M.Watanabe and Y.Nemoto, “Optimum Design of Die Casting Plunger Tip Considering Air Entrainment”, Proc. of AFC'08 (Asian Foundry Congress), pp. 455-460, Nagoya, Japan, 21 -24 May, 2008
- [72] 塩谷 實, “多変量解析概論”, 1990, 発行所: (株) 朝倉書店
- [73] B.F.J. マンリー, “多変量解析の基礎”, 1992, 発行所: (株) 培風館
- [74] G.Taguchi, S.Konishi: American supplier institute. (American Supplier Institute, 1987) 7pp.8-35.
- [75] J. Hammel, K. Hall, D. Lees, and L. Liefer, “Clinical evaluation of a desktop robotic assistant”, *Jornal of Rehabilitation Research and Development*, Vol.26, no.3(1989), pp.1-16
- [76] R. Loureiro and F. Amirabdollahian, “Upper Limb Robot Mediated Strake Therapy”, *Autonomous Robots*, Vol.15, no.1(2003), pp.35-51
- [77] M. Whittaker, “Handy-1 Robotic Aid to Eating: A Study in Social Impact”, *Proceedings of RESNA Int 92*, (1992), pp.589-594
- [78] R. Soyama, S. Ishii, A. Fukase and S. Kuroiwa, “Operating interface for meal-assistance robot 'My Spoon'”, *Proceedings of 3rd SICE System Integration Division Annual Conference*, Vol.1, (2002), pp.433-434.

- [79] Gateway Engineering Education Coalition, "The Model 5 Winsford Feeder". Cooper Union. (online), available from < <http://www.cooper.edu/engineering/projects/gateway/me/concurrent/feeders/winsford5.html> >, (accessed 2008-07-22).
- [80] Neater Solutions Ltd., "The Neater Eater", Michaelis Engineering Ltd. (online), available from < <http://www.neater.co.uk/> >, (accessed 2008-07-22).
- [81] Bestic, "Bestic - The Eating aid", Bestic. (online), available from < <http://www.bestic.se/eng/index.php> >, (accessed 2008-07-22).
- [82] J. T. Feddema, et al., "Control for Slosh-Free Motion of an Open Container, *IEEE Control Systems Magazine*, Vol.17(1997), pp.29-36
- [83] K. Yano and K. Terashima, "Robust Liquid Container Transfer Control for Complete Sloshing Suppression, *IEEE, Transactions on Control Systems Technology*, Vol.9, no.3(2001), pp.483-493
- [84] K. Yano, N. Oguro, K. Terashima, "Starting control with vibration damping by hybrid shaped approach considering time and frequency specifications", *Transactions of the Society of Instrument and Control Engineers*, vol.37, no.5(2001), pp.403-410
- [85] K. Yano and K. Terashima, "Sloshing Suppression Control of Liquid Transfer Systems Considering 3D Transfer Path, *IEEE/ASME Transactions on Mechatronics*, Vol.10, no.1(2005), pp.8-16
- [86] M. R. Barkhudarov and S. B. Chin, *Stability of a Numerical Algorithm for Gas Bubble Modelling*, "International Journal for Numerical Methods in Fluids", Vol. 19, (1994), pp.415-437
- [87] 関篤人, 森田茂隆, ダイカストの強度・信頼性に及ぼす射出の影響 "日本ダイカスト会議論文集", Vol.22, (1996)pp.18
- [88] 谷川庄司, 浅井孝一, 楊毅, 野村宏之, 加藤鋭次, ダイカストキャビティ内の溶湯圧力伝達過程に及ぼすゲート厚さの影響, "鑄造工学", Vol.75, No.8, (2003), pp.525-531
- [89] 谷川庄司, 浅井孝一, 楊毅, 野村宏之, 加藤鋭次, アルミニウム合金ダイカスト内のブローホール形態に及ぼす鑄造圧力の影響, "鑄造工学", Vol.77, No.3, (2005), pp.161-166
- [90] 加藤鋭次, 西山直之, 野村宏之, 浅井孝一, 谷川庄司, 多数個取りダイカストの充填と溶湯圧力伝達の関係, "鑄造工学", Vol.74, No.6, (2002), pp.370-376

- [91] 加藤鋭次, 瀬瀬明三, 野村宏之, 大久保真一, ダイカストにおけるスリーブ内溶湯挙動と欠陥形成, JDCA, Vol.00, No.27, (2002), pp.171-pp.177
- [92] 野村宏之, 加藤鋭次, 前田安郭, 大久保真一ダイカストスリーブ内における溶湯冷却と固相生成, “鑄造工学”, Vol.73, No.10, (2001), pp.656-661
- [93] Flow 3D, Flow Science 社, <http://flow3d.com/>
- [94] 日本流体学会編：流体力学ハンドブック（丸善）(1998), pp.376
- [95] C.W. Hirt : FSI Technical Note, **61**(2003)
- [96] 平光広太郎, 矢野賢一, 福井貴宏, 山本茂, 西戸誠志, 渡邊守, 根本泰則, 空気巻き込み量を考慮したダイカストプランジャの最適速度制御, 鑄造工学, vol.80, no.7, (2008), pp.418-423
- [97] 樋口隆英, 筒井茂義, 山村雅幸, 実数値 GA におけるシンプレックス交叉の提案, 人工知能学会誌, vol.16, no.1, (2001), pp.147-155
- [98] 小野功, 山村雅幸, 喜多一, 実数値 GA とその応用, 遺伝的アルゴリズムと最適化, 人工知能学会誌, vol.15, no.2, (2000), pp.259-266
- [99] Makoto Matsumoto and Takuji Nishimura, “Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator”, <http://saluc.engr.uconn.edu/refs/crypto/rng/matsumoto98mersenne.pdf>
- [100] Sonia Martinez, Jorge Cortes, Francesco Bullo, “Motion Coordination with Distributed Information” IEEE Control System Magazine, (2007), pp.75～pp.87, 27, Aug.
- [101] 寺嶋一彦; “鑄造業における自動注湯・溶湯搬送の現状”, 素形材, VOL. 39, NO. 6, pp. 1-8, 1998
- [102] C.W.Hirt, B.D.Nichols, *Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries*, “Journal of Computational Physics”, 39, 6, (1981), pp.201,
- [103] C.W.Hirt, J.M.Sicilian, *A Porosity Technique for the Definition of Obstacles in Rectangular Cell Meshes*, “Proc. of 4 th International Conference on Ship Hydrodynamics”, (1985),

付録A CFDシミュレータ

A.1 基礎式

基礎式は、質量保存式 (A.1)、運動量保存式、液体率関数の保存式およびエネルギー保存式により構成される。未知数は圧力 P 、流速 u, v, w 、内部エネルギー I 、流体率 F および固体温度 T_w である。 V_F はセル体積率、 A_x, A_y, A_z は面開口率を表し、これらのボロシティ関数を用いて曲面障害物環境が定義される。 ρ は液体の密度、 $RDIF$ は乱流乱数項、 $RSOR$ は質量湧出項、 G_x, G_y, G_z は加速度項、 f_x, f_y, f_z は粘性項、 b_x, b_y, b_z は流動圧損項、 K は抗力係数、 I はマクロ的な混合内部エネルギー、 $RIDIF$ は乱流拡散項、 $TDIF$ は熱伝導項、 $RISO$ は熱伝達項も含まれるエネルギー湧出項、 $FSOR$ は式の $RSOR$ に相当する。差分法はコントロール・ボリューム法 (有限差分法) のオイラーメッシュに基づく。

$$F_f \frac{\delta \rho}{\delta t} + \frac{\delta}{\delta x}(\rho u A_x) + \frac{\delta}{\delta y}(\rho v A_y) + \frac{\delta}{\delta z}(\rho w A_z) = RDIF + RSOR \quad (A.1)$$

$$\frac{\delta u}{\delta t} + \frac{1}{V_F} \left(u A_x \frac{\delta u}{\delta x} + v A_y \frac{\delta v}{\delta y} + w A_z \frac{\delta w}{\delta z} \right) = -\frac{1}{\rho} \frac{\delta p}{\delta x} + G_x + f_x - b_x - \frac{RSOR}{\rho V_F} u - K u \quad (A.2)$$

$$\frac{\delta v}{\delta t} + \frac{1}{V_F} \left(u A_x \frac{\delta u}{\delta x} + v A_y \frac{\delta v}{\delta y} + w A_z \frac{\delta w}{\delta z} \right) = -\frac{1}{\rho} \frac{\delta p}{\delta y} + G_y + f_y - b_y - \frac{RSOR}{\rho V_F} v - K v \quad (A.3)$$

$$\frac{\delta w}{\delta t} + \frac{1}{V_F} \left(u A_x \frac{\delta u}{\delta x} + v A_y \frac{\delta v}{\delta y} + w A_z \frac{\delta w}{\delta z} \right) = -\frac{1}{\rho} \frac{\delta p}{\delta z} + G_z + f_z - b_z - \frac{RSOR}{\rho V_F} w - K w \quad (A.4)$$

$$\frac{\delta F}{\delta t} + \frac{1}{V_F} \left(\frac{\delta}{\delta x} F A_x u + \frac{\delta}{\delta y} F A_y v + \frac{\delta}{\delta z} F A_z w \right) = FDIF + FSOR \quad (A.5)$$

A.2 VOF 法

VOF 法^[102]は計算格子に存在する液体率を VOF 関数で管理する。この VOF 関数は、液体で満たされた計算セルを 1、存在しない計算セルを 0、液体が部分的に存在する計算セルをその体積占有率に応じて 0 ~ 1 の間で表現する。なお、各接点における気液相の密度および粘性係数は VOF 関数を用いて式 (A.6)、式 (A.7) のように表すことができる。

$$\rho = \rho_{Liq}\emptyset + \rho_{Gas}(1 - \emptyset) \quad (A.6)$$

$$\mu = \mu_{Liq}\emptyset + \mu_{Gas}(1 - \emptyset) \quad (A.7)$$

ここで Liq は液相, Gas は気相を示す.

A.3 固液相流動

溶湯の凝固に伴い, 固相と液相が同時に存在する固液相状態となる. 低い固相率での固相の結晶はまばらであり, 液体中に自由に浮かぶ. そのときの固液相状態は式 (A.8) により, 1 液体の粘性変化として近似される.

$$\mu = \mu_0 \left(1 - \frac{F_s}{FSCR}\right)^{-1.55} \quad (A.8)$$

ここで μ_0 は液相の粘度, F_s は局所の固相率, $FSCR$ は流動限界固相率である. この粘性の変化は式 (A.2)~(A.4) の右辺粘性項で反映される. 凝固が進み固相率がある基準値 ($FSCR$) に至ると, 液体の粘性はそのときの値で保持される. そしてその固液相状態における流動変化に対する簡易化されたモデルとして, ダルシータイプの抗力モデルが使用される. これはマッシー (かゆ状) 凝固における透過媒体の抗力の概念を用いて構成され, 抗力係数 K は式 (A.9) で表現される.

$$K = TSDRG \frac{F_s^2}{(1 - F_s)^3} \quad (A.9)$$

ここで $TSDRG$ は凝固抗力係数である. そして $F_s = FSCR$ で流動不可能となる. $FSCO$ と $FSCR$ の関係を次式に示す.

$$0.0 \leq FSCO \leq FSCR \leq 1.0 \quad (A.10)$$

抗力係数 K の変化は式 (A.2)~(A.4) の右辺浸透性媒体項で反映される.

A.4 メッシュ生成法

本 CFD シミュレータソフトは, 生成が容易な直交格子に強力な形状表現能力を合体させた, 自由度の高い *FAVOR* 法 (*Fractional Area Volume Obstacle Representation*)^[103] と呼ばれる格子生成手法を採用しており, BFC や FEM のように物理形状に表面メッシュを作成する必要がなく, 空間格子と物体表面形状は完全に独立して定義することができる.

*FAVOR*法は直交格子をベースに，セルの一部をカットして滑らかな形状を表現することが可能である．直交格子と障害物形状が定義されるとプリプロセッサが障害物で塞がれた計算セル壁面の開口率と体積率を全ての計算セルに割り振り，空間格子状に形状が精度よく再現される．

付 録 B 溢流抑制制御の搬送軌道に対する速度カーブの導出過程

溢流抑制制御の搬送軌道に対する速度カーブの導出過程について述べる. 用いる 7 次関数の式は以下ようになる.

$$f(t) = \sum_{i=0}^n a_i t^i \quad (\text{B.1})$$

$$f'(t) = \sum_{i=0}^{n-1} (a_{i+1} t^i)(i+1) \quad (\text{B.2})$$

$$f''(t) = \sum_{i=0}^{n-2} (a_{i+2} t^i)(i^2 + 3i + 2) \quad (\text{B.3})$$

ここで $n=7$, t は時間, a_i ($i = 0 \sim 7$) は定数, f は速度を表している.

この速度カーブの制約条件として, 初期条件をそれぞれ $f(0) = 0.0[\text{m/s}]$, $f'(0) = 0.0[\text{m/s}^2]$,

$f''(0) = 0.0[\text{m/s}^3]$ とし, 加速終了時間 t_1 における条件をそれぞれ $f(t_1) = V_{\text{target}}[\text{m/s}]$, $f'(t_1) = 0.0[\text{m/s}^2]$, $f''(t_1) = 0.0[\text{m/s}^3]$ とする.

初期条件から式 (B.4) を得ることができる.

$$a_0 = a_1 = a_2 = 0 \quad (\text{B.4})$$

また, 加速終了時間の条件から式 (B.5)～式 (B.7) を得ることができる.

$$V_{\text{target}} = a_7 t_1^7 + a_6 t_1^6 + a_5 t_1^5 + a_4 t_1^4 + a_3 t_1^3 \quad (\text{B.5})$$

$$0 = 7a_7 t_1^6 + 6a_6 t_1^5 + 5a_5 t_1^4 + 4a_4 t_1^3 + 3a_3 t_1^2 \quad (\text{B.6})$$

$$0 = 42a_7 t_1^5 + 30a_6 t_1^4 + 20a_5 t_1^3 + 12a_4 t_1^2 + 6a_3 t_1 \quad (\text{B.7})$$

ここで, 連立方程式の数は 3 つ存在する. つまり, 変数を V_{target} , t_1 , a_7 および a_6 の 4 つを変数と考えれば, 残りの a_5, a_4, a_3 はこの連立方程式を用いることで決定される. したがって, a_5, a_4, a_3 について解く必要がある.

まず, 式 (B.5)～式 (B.7) を式 (B.8) のような行列式に書き直す. ただし, 式 (B.6), 式 (B.7) の両辺にはそれぞれ t_1, t_1^2 を事前に掛けておく.

$$\begin{bmatrix} 1 & -1 & -1 \\ 0 & -7 & -6 \\ 0 & -42 & -30 \end{bmatrix} \begin{bmatrix} V_{\text{target}} \\ a_7 t_1^7 \\ a_6 t_1^6 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 5 & 4 & 3 \\ 20 & 12 & 6 \end{bmatrix} \begin{bmatrix} a_5 t_1^5 \\ a_4 t_1^4 \\ a_3 t_1^3 \end{bmatrix} \quad (\text{B.8})$$

式 (B.8) をガウス消去法によって計算を行っていく。まず、2 行目から 1 行目を 5 倍したものを引き、3 行目から 1 行目を 20 倍したものを引くと式 (B.9) のようになる。

$$\begin{bmatrix} 1 & -1 & -1 \\ -5 & -2 & -1 \\ -20 & -22 & -10 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & -1 & -2 \\ 0 & -8 & -14 \end{bmatrix} \quad (\text{B.9})$$

次に、3 行目から 2 行目を 8 倍したものを引くと式 (B.10) を得る。

$$\begin{bmatrix} 1 & -1 & -1 \\ -5 & -2 & -1 \\ 20 & -6 & -2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & -1 & -2 \\ 0 & 0 & 2 \end{bmatrix} \quad (\text{B.10})$$

次に、2 行目から 3 行目を 1 倍したものを足し、3 行目を 2 で割ると式 (B.11) を得る。

$$\begin{bmatrix} 1 & -1 & -1 \\ 15 & -8 & -3 \\ 10 & -3 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B.11})$$

そして、1 行目から 2 行目を 1 倍した物を足す、1 行目から 3 行目を引く、2 行目を -1 倍すると式 (B.12) を得る。

$$\begin{bmatrix} 6 & -6 & -3 \\ -15 & 8 & 3 \\ 10 & -3 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B.12})$$

最終的に、式 (B.13) となる。

$$\begin{bmatrix} 6 & -6 & -3 \\ -15 & 8 & 3 \\ 10 & -3 & -1 \end{bmatrix} \begin{bmatrix} V_{\text{target}} \\ a_7 t_1^7 \\ a_6 t_1^6 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_5 t_1^5 \\ a_4 t_1^4 \\ a_3 t_1^3 \end{bmatrix} \quad (\text{B.13})$$

従って式 (B.13) より a_5, a_4, a_3 は式 (B.14)~式 (B.16) のようになる。

$$a_5 = \frac{6V_{\text{target}} - 6a_7 t_1^7 - 3a_6 t_1^6}{t_1^5} \quad (\text{B.14})$$

$$a_4 = \frac{-15V_{\text{target}} + 8a_7 t_1^7 + 3a_6 t_1^6}{t_1^4} \quad (\text{B.15})$$

$$a_3 = \frac{10V_{\text{target}} - 3a_7 t_1^7 - a_6 t_1^6}{t_1^3} \quad (\text{B.16})$$

付 録 C CFDシミュレータを援用した 自動注湯の最適傾動速度の導出

C.1 緒言

現在、自動車産業をはじめとする様々な製品の製造現場でアルミニウム合金を用いた重力鋳造が行われている。アルミ重力鋳造は金型を用いるため寸法精度が高く、また製造プロセスの機械化・自動化が容易であり、砂型鋳物に対して生産性が高く、量産に適しているなど多くの利点を持つ。

このアルミ重力鋳造における製品欠陥の一つとして、ブローホールやピンホールが挙げられる。これらの欠陥防止策としては、型からのガス抜け改善、湯口方案の変更、鋳型堰部での流速の安定化などの改善があげられる。ガス抜け改善と湯口方案変更には金型の新規制作や改良にコストがかかるため、本研究では、注湯速度を最適に制御することで、鋳型堰部での流速を安定させ、製品欠陥率を低減することを目的とする。

現在のアルミ重力鋳造の製造現場では、鋳型へ溶湯を注ぎ込む傾動式自動注湯機が多く用いられ、作業者が経験と勘を頼りに複数回の試験を繰り返すことにより手動で注湯速度を決定している。しかしながら、試行錯誤で最適な多段階の注湯速度を求めるのは、その組み合わせ数が莫大となり、現実的には困難である。そのため、高品質な鋳物を実現する適切な鋳込みが行われているとは言えず、製品不良によって歩留まりの低下を招いている。さらに、このような試行作業により生産準備期間の伸張やコストの増大も大きな問題となっている。加えて、従来研究ではシミュレーションの計算時間の関係から、砂型鋳造や重力鋳造へのCFDシミュレータを援用した最適化技術の確立が遅れているのが現状である。

そこで本研究では、CFDシミュレータを援用し、湯口カップ内を溶湯ですばやく満たし、液位を高く一定に保つ液面レベル制御を行うことで、鋳型堰部での流速を安定させ、アルミ重力鋳造におけるピンホールやブローホールなどの欠陥低減を実現する最適注湯速度入力を導出することを目的とする。自動注湯機には、多くの鋳造工場で普及している多段階速度入力タイプのものを対象とし、対象鋳型にはインテークマニホールドの実製品形状を用いる。また、注湯動作における取鍋傾動速度の導出には遺伝的アルゴリズムを適用し、最適化を行う。本研究の有効性は、流体解析シミュレーションおよび実機テストにより示す。

C.2 実験装置

C.2.1 自動注湯機の構成と注湯指令入力方法

本研究で用いる自動注湯機の概観を Fig. C.1 に示す。株式会社メイチュウ社製傾動式自動注湯機で、ラドルの前後方向の位置と傾動軸の2自由度を持つ。傾動実測値の表示および外部への出力には対応していない。ラドルへの溶湯の補給は電気炉から直接溶湯をラドルですくい上げる方式で、すくい上げた後、ラドル後部の切り欠きから溶湯を落とし湯量を調節する。対象製品形状には、インテークマニホールドを用いる。

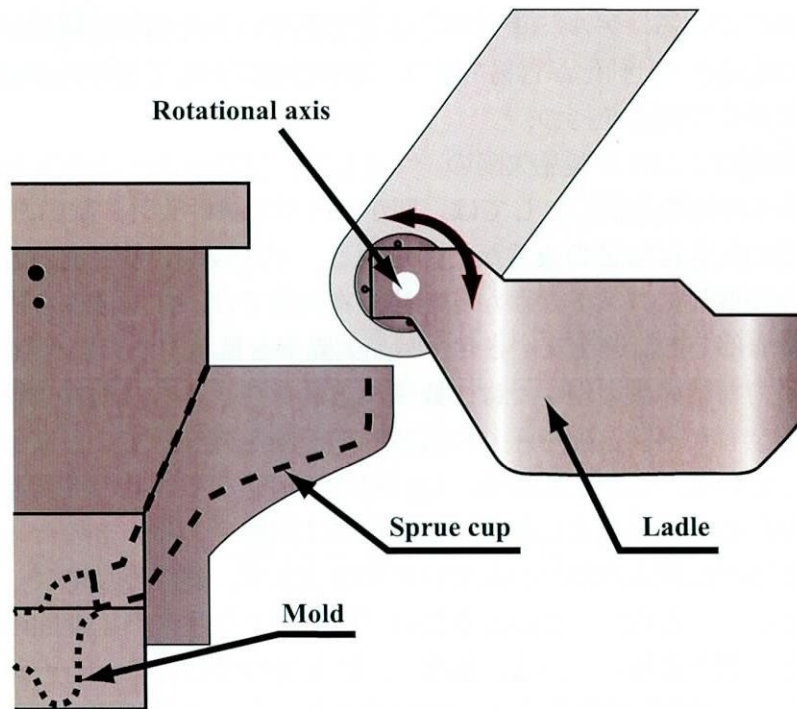


Fig. C.1 Overview of the automatic pouring machine

本研究で用いる注湯機では通常、百分率で表される傾動角速度 v と速度切り替え角度 θ をティーチングペンダントで設定し、注湯速度を決定する。設定は傾動角速度4段階、切り替え角度3段階の計7変数使用可能である。傾動角速度の目標値は Fig. C.2 に示すように角加速度 $\alpha[\text{deg/s}^2]$ で立ちあがる台形状に与えられる。ここで、初期角度と傾動終了角度は、鑄込み量により事前に決定される。傾動角速度設定が百分率で表されることから、各段階の傾動角速度 $V_n[\text{deg/s}]$ は最大角速度 V_{\max} を用いて次式で示される。

$$V_n = \frac{V_n}{100} V_{\max} \quad (\text{C.1})$$

ここで、角加速度 $\alpha[\text{deg/s}^2]$ と最大角速度 V_{\max} は未知パラメータとなるため、パラメータ同定を行う必要がある。

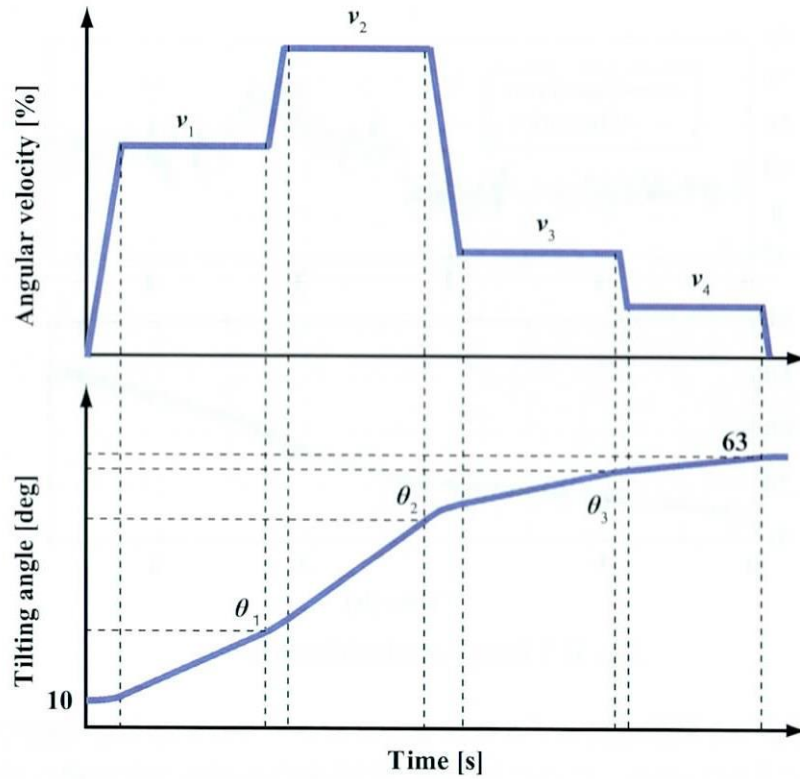


Fig. C.2 Tilting velocity of pouring machine

Table C.1 Setting of the tilting input

	Switching angle [deg]	Pouring speed [%]
$n = 1$	22	10
$n = 2$	32	30
$n = 3$	42	50
$n = 4$	—	30

C.2.2 注湯機の動作解析と傾動速度モデル

本研究で実験機として用いる注湯機は、傾動速度や角度などの表示・出力ができないため、設定入力に対してどのような動作をしているかはわからない。CFDシミュレータを用いて溶湯挙動の評価を行うに当たり、シミュレーション上のラドル傾動動作と実験機との整合性を図ることは不可欠である。そこで、2台のカメラを用いて、その傾動動作を撮影し、鋳造工場内で実際に使用されている注湯機の動作解析を行った。計測に用いた注湯速度設定を Table C.1 に示す。

撮影されたデータから得られた傾動角速度データを用いて、本プラントの最大傾動角速度と角加速度を導出した。結果、最大傾動角度と角加速度はそれぞれ $V_{\max} = 51.9[\text{deg/s}]$, $\alpha = 200[\text{deg/s}^2]$ であった。また、動作計測結果と傾動速度モデルによるシミュレーション結果を Fig. C.3 に示す。

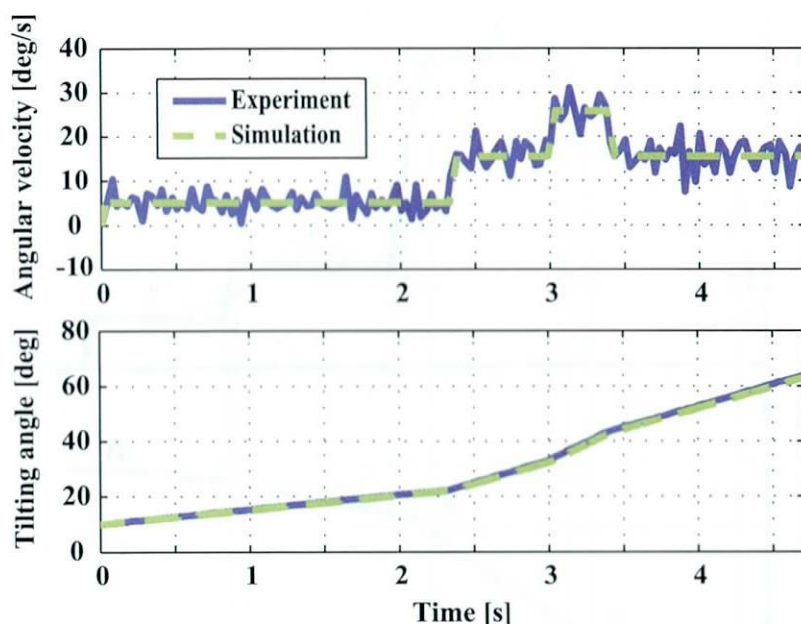


Fig. C.3 Result of identification

Fig. C.3の結果より、傾動速度モデルによるシミュレーション結果と実験結果が良好に一致していることがわかる。このことから、傾動速度モデルが注湯機の挙動を精度良く表現できていることが確認された。

C.3 CFDシミュレータの流動評価

本研究では、流体解析ソフトウェアであるFlowScience社製の*FLOW-3D*を用いる。このシミュレータは非圧縮流れから自由曲面を伴う流れ、圧縮性を考慮した流れ、さらに凝固を伴う流れまで広範囲な流れを扱うことが可能な差分法を用いた3次元流体計算プログラムである。このシミュレータの妥当性は、多くの文献においてすでに示されている^{[65][93][96]}。

本研究では、対象流体をAC2Bとする。鋳込み量は $1.863 \times 10^{-3} [\text{m}^3]$ 、製品部体積は $1.429 \times 10^{-3} [\text{m}^3]$ である。解析におけるAC2Bの物性値はTable C.2のように設定した。また、シミュレーション領域におけるメッシュの概要図をFig. C.4に、メッシュパラメータはTable C.3に示す。

Table C.2 Fluid parameters of AC2B

Fluid parameters	AC2B
Density [kg/m^3]	2550
Viscosity [$\text{Pa}\cdot\text{s}$]	0.00125
Temperature of the Fluid [K]	993

Table C.3 Mesh parameters of pouring machine

Mesh block	Cell size [m]	number of cell
X-direction	0.0040 ~ 0.080	108
Y-direction	0.0050	54
Z-direction	0.0040 ~ 0.080	65
Total number of cell		379080

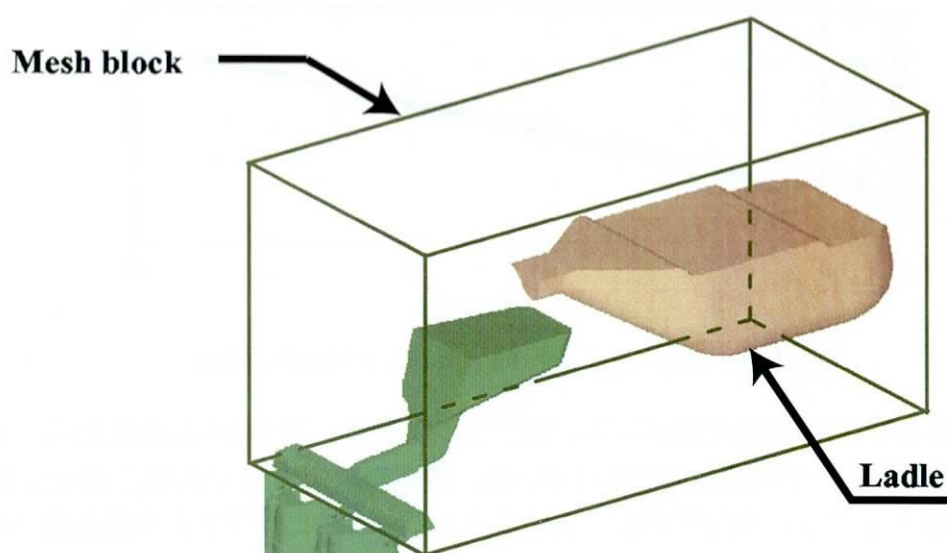


Fig. C.4 Mesh setting of pouring machine

Fig. C.1 に示すように、注湯機と鋳型周辺は大変入り組んでおり、溶湯の挙動を観測できる点が限られる。観測できる視点が、鋳型湯口付近の真横からの視点と、湯口上方からの視点のみに限られるため、この2点での溶湯挙動比較とモデルパラメータ同定を行う。なお、傾動入力には試行錯誤によって設定した注湯速度設定を使用する。その際の注湯速度設定を Table C.4 に、傾動角度と角速度、そして角加速度を Fig. C.5 に示す。

Table C.4 Input setting for the parameter identification

	Switching angle [deg]	Pouring speed [%]
$n = 1$	22	20
$n = 2$	32	10
$n = 3$	42	10
$n = 4$	—	10

Fig. C.6 に、注湯時における実験とシミュレーションの流線の比較結果を示す。

結果として、本シミュレータを用いて、実機による注湯中の溶湯の流出挙動を良好に再現できていることが確認できる。ここで図中破線は湯口上端を示す。

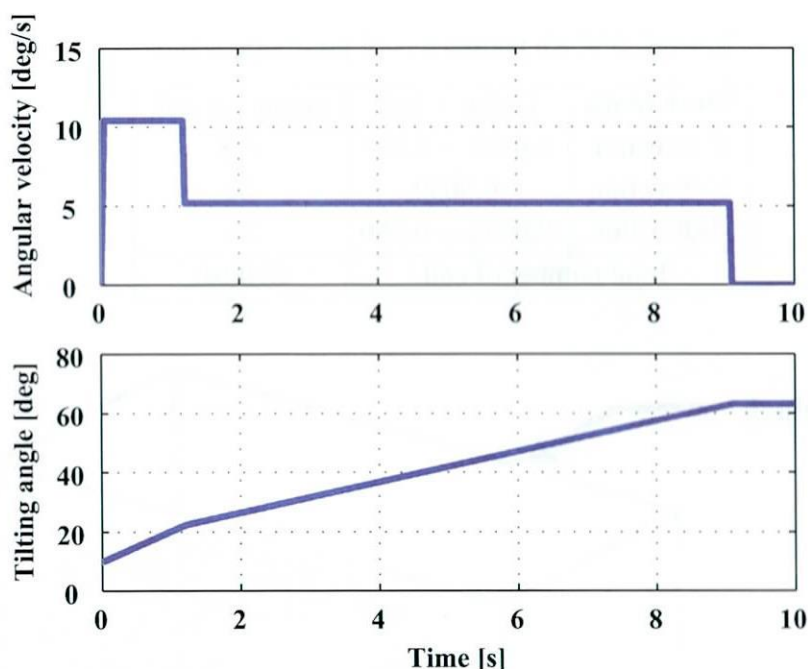


Fig. C.5 Tilting input for the parameter identification

次に、湯口内の溶湯挙動を解析する。実プラントでは、スラグの除去を目的として溶湯フィルタ（金網）が湯道に設置されている。金網の線太さは0.5[mm]、線の本数は縦50、横55本である。本研究では、CFDシミュレータでこのフィルタを再現するために、多孔質バッフル（以下、バッフルと呼称する）を用いた。

バッフルの設定パラメータとして、空隙率 b_p 、線形速度下降係数 b_l 、2次速度下降係数 b_q を与える。 b_l と b_q はEq. (C.2)に示すバッフル流損の式で定義される。

$$B = \frac{1}{L} (b_l u + 0.5 b_q u |u|) \quad (\text{C.2})$$

ここで、 B はバッフル流損、 u はバッフル内での流速、 L は流損が生じる長さを示す。金網の空隙率は、金属線とその隙間の面積比から計算することができる。バッフル流損は線形速度下降が支配的であるため、2次速度下降係数を $b_q = 0$ で固定し、線形速度下降係数 b_l の探索を行った結果をTable C.5に示す。なお、探索範囲は $b_l = 0.00 \sim 1.50$ とし、0.05刻みで探索を行った。バッフル流損を考慮したシミュレーション結果をFig. C.7に示す。

Table C.5 Parameters of porous baffle

Porosity [-]	0.655
Linear loss coefficient : b_l [-]	0.90
Quadratic loss coefficient : b_q [-]	0.00

注湯実験結果と比較すると、良好に湯口内の溶湯挙動を再現できていることがわかる。

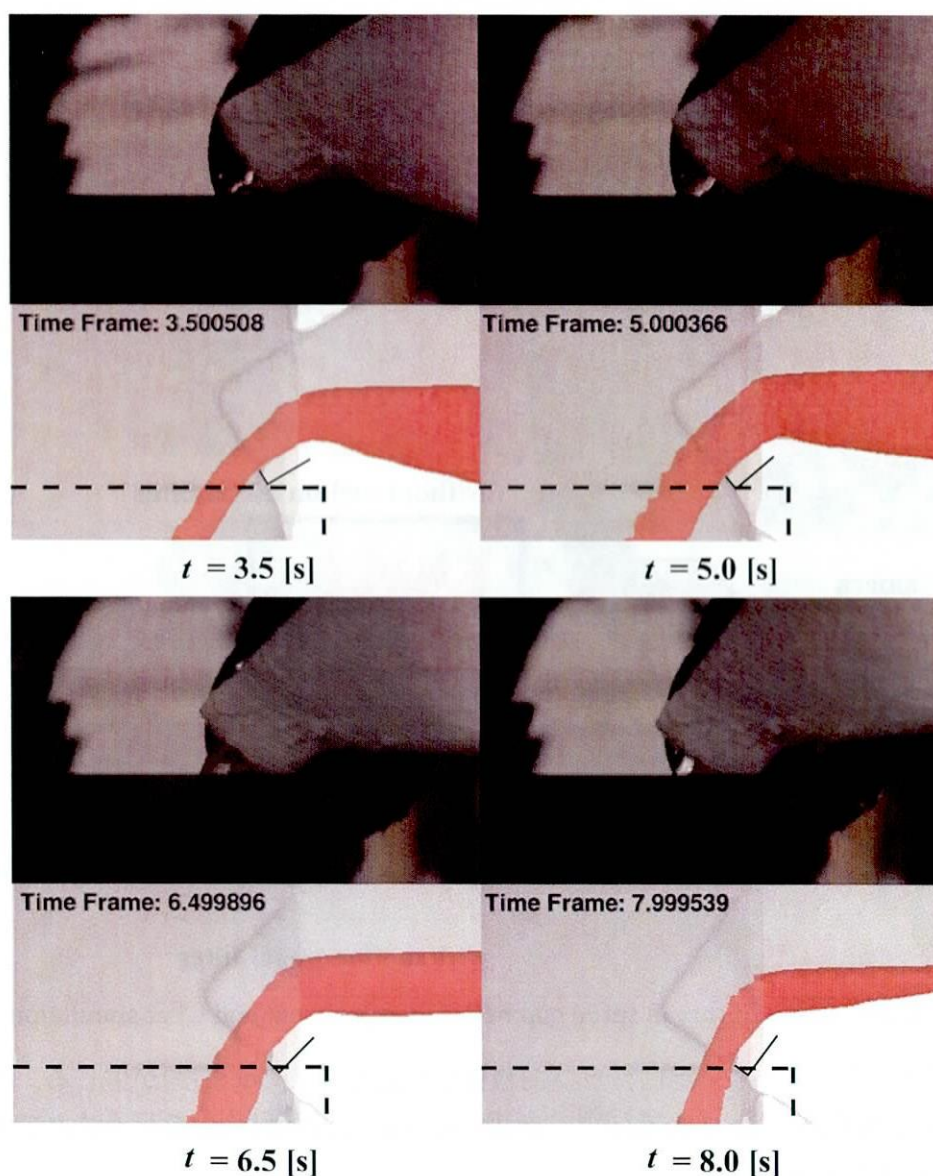


Fig. C.6 Comparison of flow line between experiment and CFD simulation

C.4 遺伝的アルゴリズムを用いた最適注湯入力の変出

C.4.1 液面レベル制御による製品欠陥低減

湯口カップ内を溶湯ですばやく満たし、液位を高く一定に保つ液面レベル制御を行うことで、鑄型堰部での流速を一定にさせ、

- 製品部充填時間の短縮
- 乱流の発生防止
- 空気などの巻き込みの防止

を実現し、製品欠陥を低減した高品質な鑄物を製造できるとされている^[101]。しかしながら、作業者の試行錯誤により注湯速度を決定した場合、不適切な速度設定が原因で

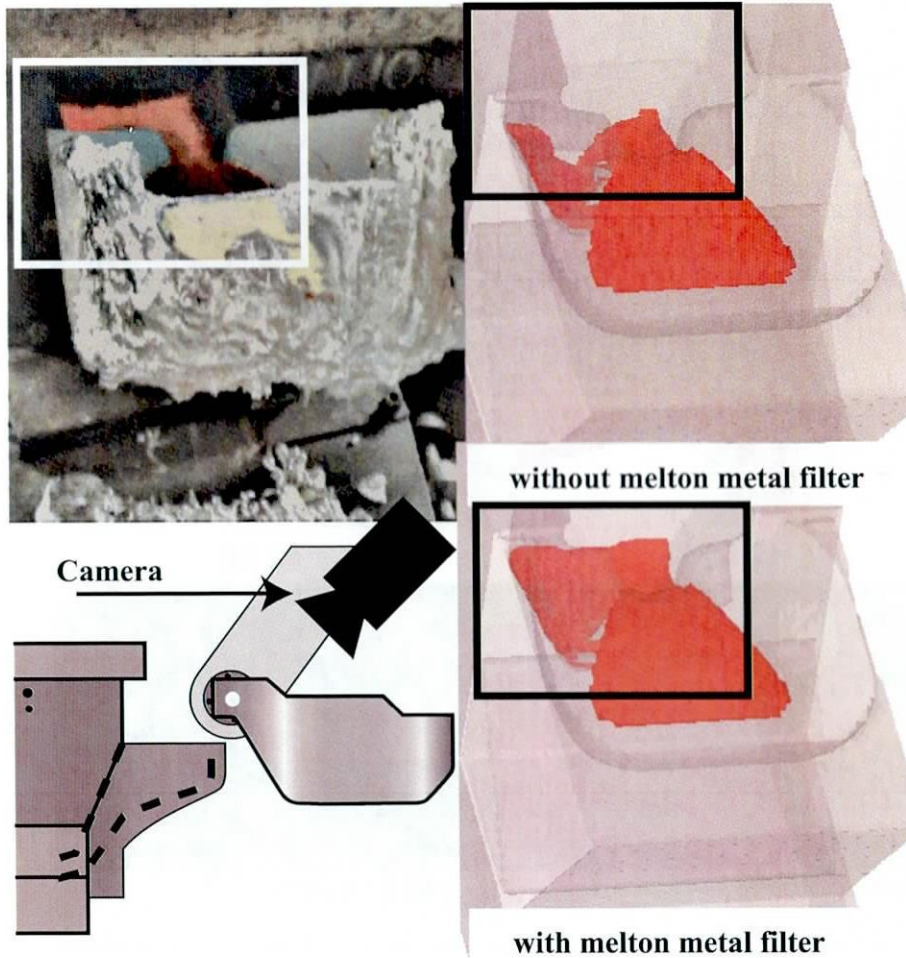


Fig. C.7 Comparison of flux in sprue cup between experiment and CFD simulation

鋳型から溶湯が溢れ出たり、湯口カップ内液位が安定しない等の問題が生じる。そこで本研究では、遺伝的アルゴリズム^{[66][54]}を用いた注湯入力最適化を行うことで液面レベル制御を実現する。

C.4.2 最適化問題の定式化

注湯機への設定入力である切り替え角度3段階、注湯速度4段階の計7個の設定パラメータを変数とし、実機制約内で最適な傾動速度パターンを導出する。湯口カップをすばやく満たし、液面を一定レベルに安定化させるために、傾動終了時間を評価関数とし、湯口内液位を制約条件としてEq. (C.3)に示す最適化問題に定式化した。傾動終了時間を評価関数とすることで注湯時間が短縮され、更に液面レベルを制約条件とすることで液面レベル制御が可能となる。

$$\text{minimize} : J = t_p + J_p \quad (\text{C.3})$$

$$h \geq 0.025 \quad (\text{C.4})$$

ここで, t_p は傾動終了時間, J_p は Eq. (C.5) に示すペナルティ関数, h は湯口上面から溶湯までの変位を示す.

$$J_p = w_1 + w_2 \quad (\text{C.5})$$

Eq. (C.5) において, w_1 は注湯の際に湯口上面から湯口内液位までの変位が 0.025[m] を下回った場合 ($h < 0.025$) に, w_2 は注湯機で再現できない個体が生成された場合にペナルティ項 $w_1, w_2 = 10^2$ が課せられる.

C.4.3 最適化計算結果と湯流れシミュレーション

Table C.6 に示す計算パラメータを用い, 遺伝的アルゴリズムによる最適化を行った. 最適化計算の結果を Fig. C.8 に示す. 最適化に要する時間は Intel Core2 Quad CPU 搭載のパソコンで 48 時間を要した.

Table C.6 Parameters for GA

Number of variable	7
Number of population	10
Number of elite preservation	1
Mutation fraction	0.01
Crossover fraction	0.80

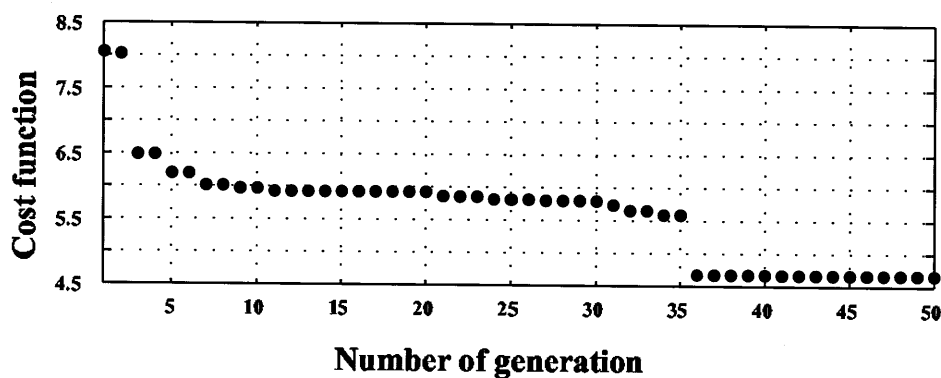


Fig. C.8 Cost function

Fig. C.8 より, 第 41 世代までに解が収束し, 最適化が行われたことを確認できる. その時, 評価値は $J = 4.668$ となり, 最適解は Table C.7 に示すパラメータを得た. 最適パラメータから得られた傾動角度・速度を Fig. C.9 に示す. また, Fig. C.10 に液面レベル制御を考慮した場合としない場合の各々の時間における高さの変化を示す.

結果として, 制御なしの場合では目標液面レベルまで到達しておらず, 十分な液面を保つことができていないことがわかる. それに対し, 注湯制御入力の最適化を行うことで, すばやく湯口カップを満たす良好な液面レベル制御を実現できていることがわかる.

Table C.7 Input setting of optimization result

	Switching angle [deg]	Pouring speed [%]
$n = 1$	12	54
$n = 2$	15	78
$n = 3$	39	86
$n = 4$	—	10

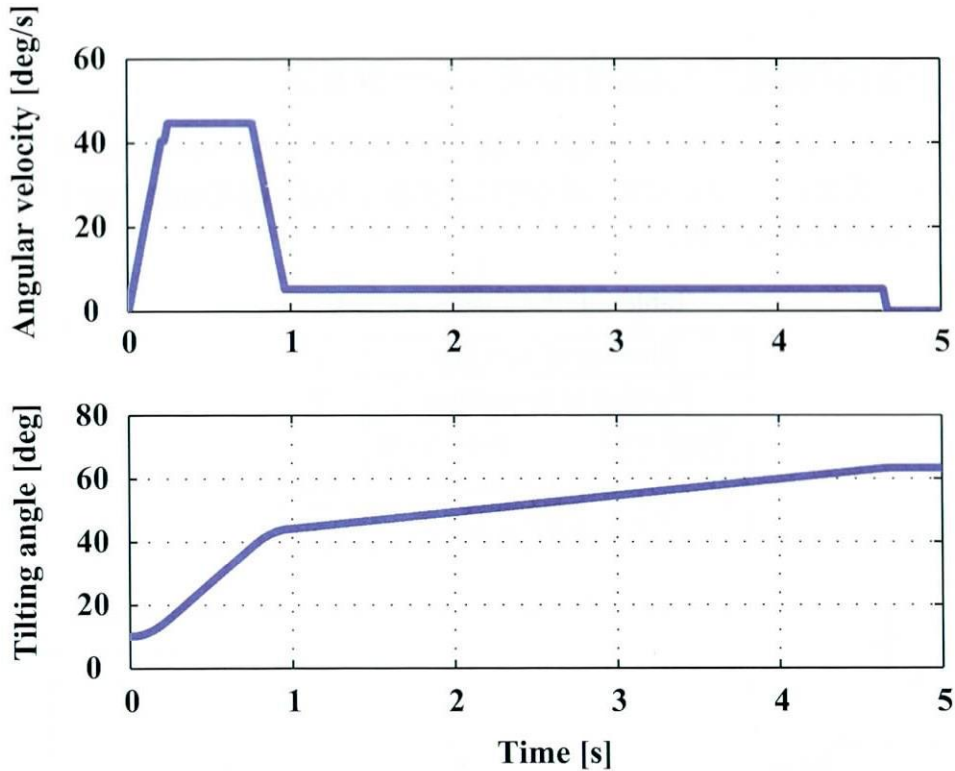


Fig. C.9 Tilting input of optimization result

C.5 製品部を考慮した解析結果の評価

製品部の湯流れシミュレーション結果を Fig. C.11 に示す. このとき液面レベル制御によって生じる空気巻き込みの度合いを調べるために空気連行モデルを使用して解析を行った. 切削加工面の空気連行の様子を Fig. C.12 に示す.

Fig. C.12 と Table C.8 より, これまでの入力と比較し, 最適化を行うことで注湯時間の短縮を実現するだけでなく, 液面レベル制御は空気連行低減にも有効であることが確認できる.

C.6 最適速度入力を用いた注湯制御実験

求めた最適速度入力を実機に適用し, 注湯実験を行った. 最適解の比較対象には, Table C.4 に示す入力を用いた. 鑄造試行回数は 11 回とし, 湯口, 方案, 押し湯部分を

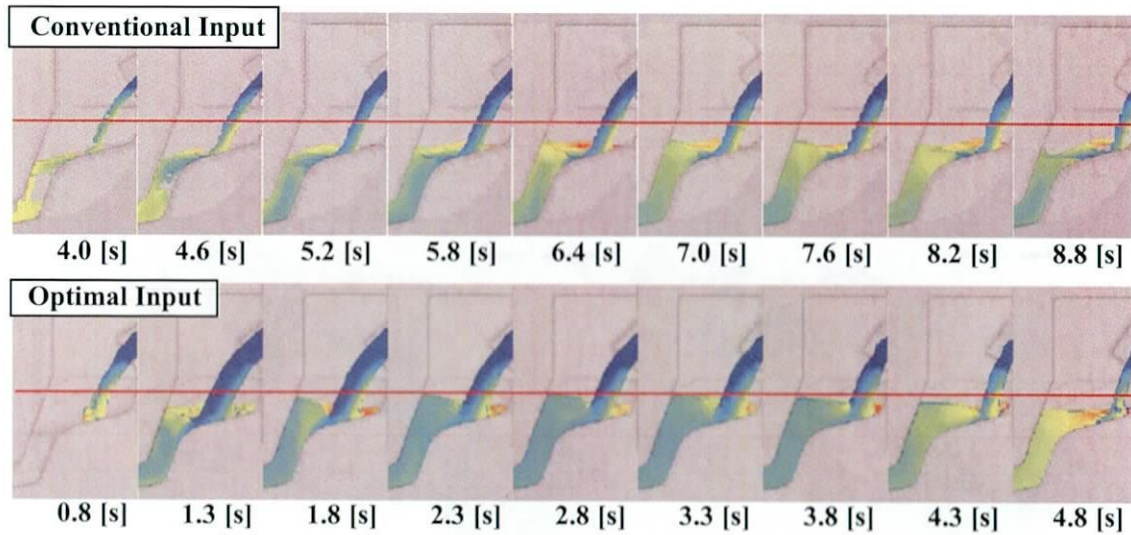


Fig. C.10 Comparison of surface height between optimal input and conventional input

Table C.8 Simulation results of air entrainment

	Conventional input	Optimum input
Cost function J	9.1	4.7
Filling Time [s]	9.2	5.3
Air entrainment	4.983	4.348

除いた製品部切削加工後の加工面に現れる欠陥の発生率を評価した。Fig. C.13 に各入力パラメータを用いた実験結果，Fig. C.14 にそれぞれの欠陥発生率を示す。

結果として，Fig. C.13 および Fig. C.14 に示すように，最適注湯速度入力を用いることで，製品欠陥発生率を 55% から 9% まで低減させることが可能となった。これにより，提案した手法を用いて導出された最適入力パラメータは製品欠陥発生率の低減に対して有効であることがわかった。

C.7 結言

本研究では，流体解析シミュレーションにおける計算時間短縮を可能とする解析手法を提案し，アルミ重力鋳造におけるブローホールやピンホールなどの欠陥発生低減を目的とした注湯制御入力の最適化を行った。遺伝的アルゴリズムを用いた注湯制御入力の最適化を行った結果，最適注湯制御入力は液面レベル制御を実現させ，実機注湯実験によって製品欠陥低減に対して有効であることがわかった。

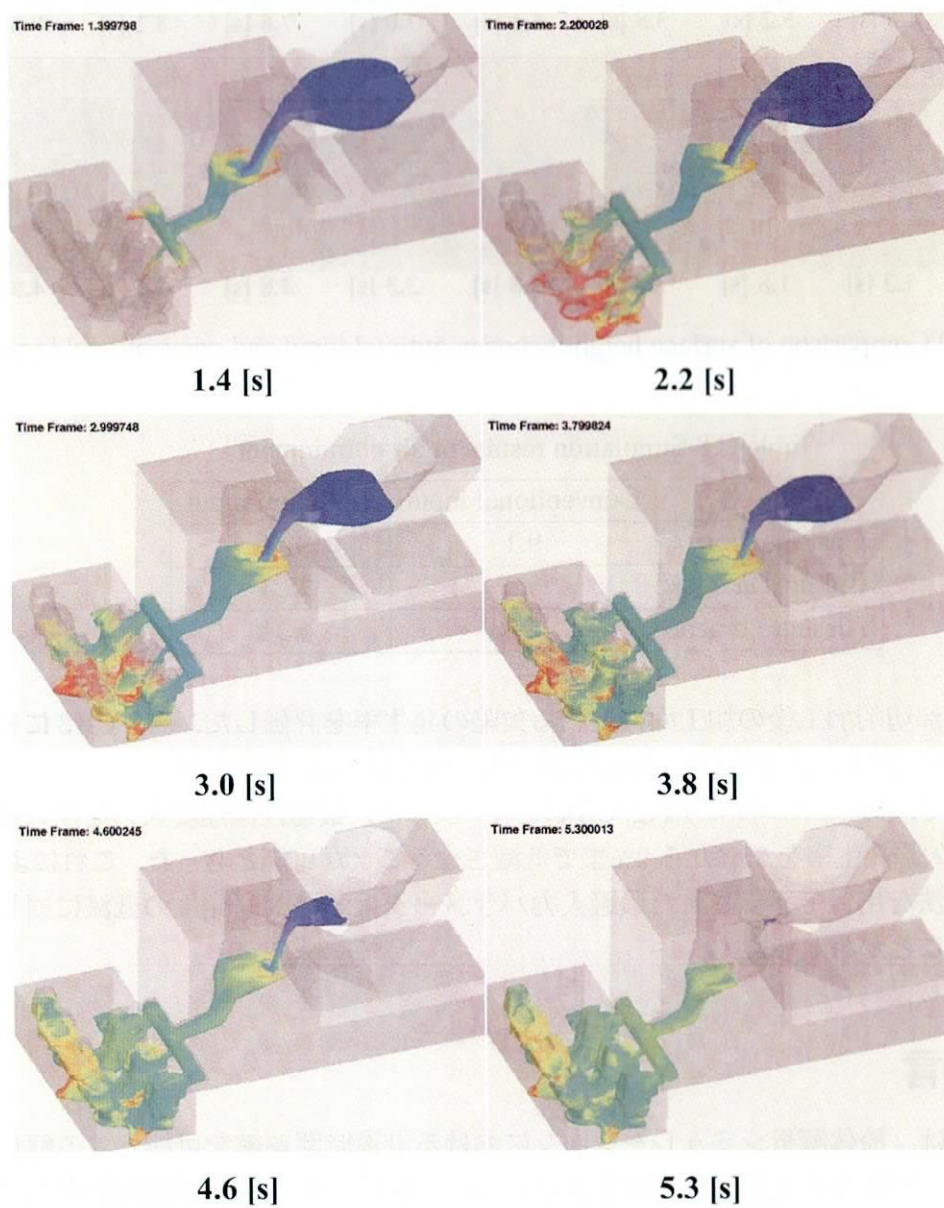


Fig. C.11 Simulation result with optimum tilting velocity

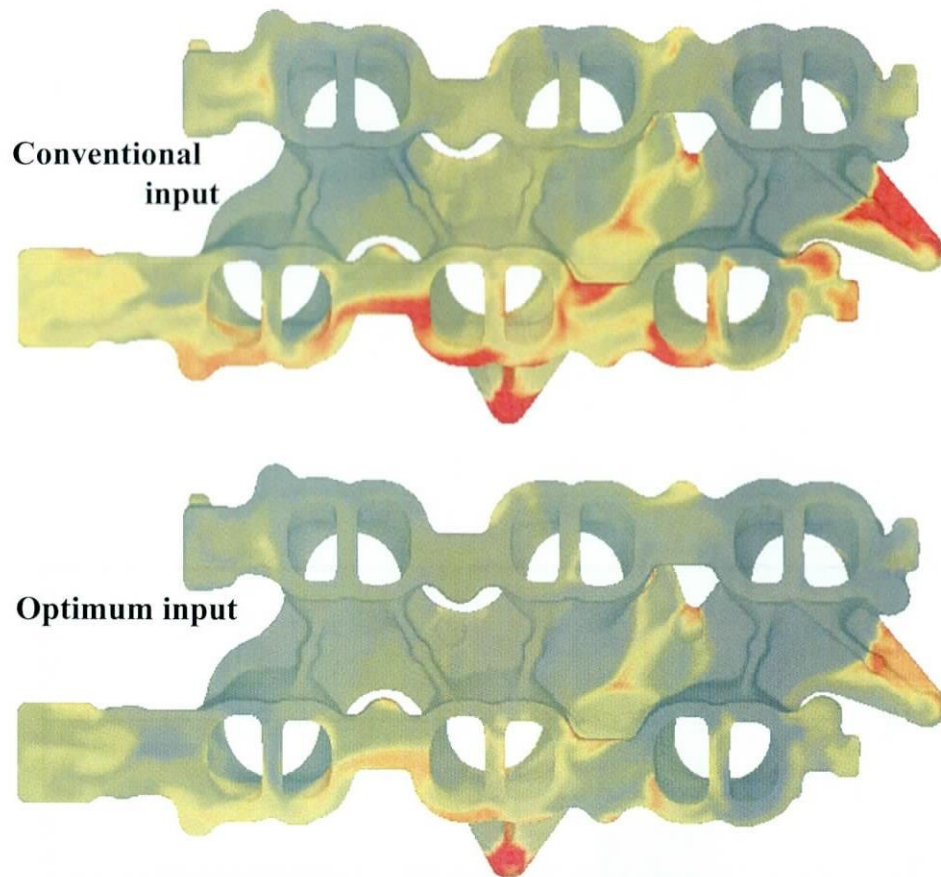


Fig. C.12 Air entrainment at machining surface

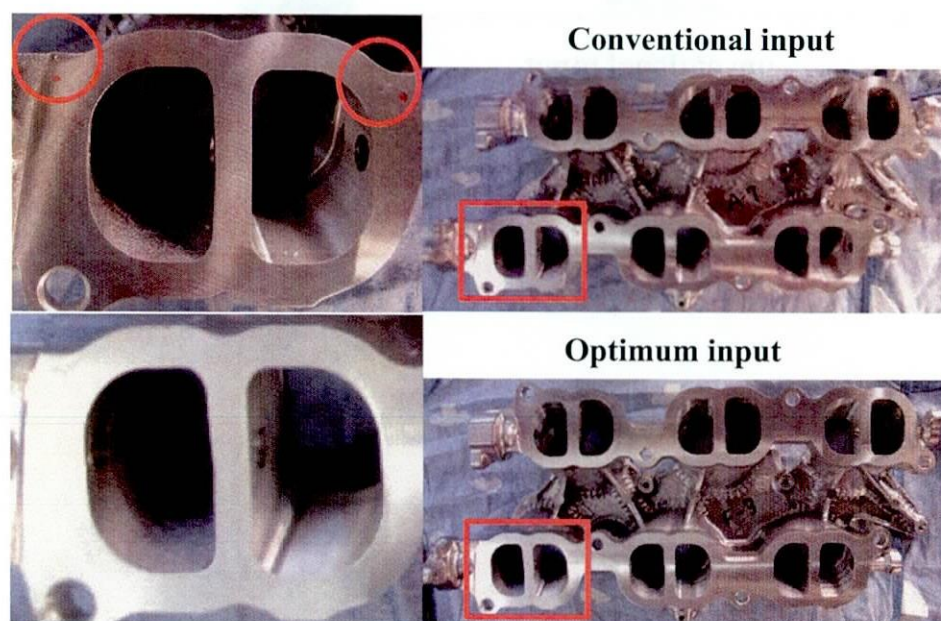


Fig. C.13 Experimental results

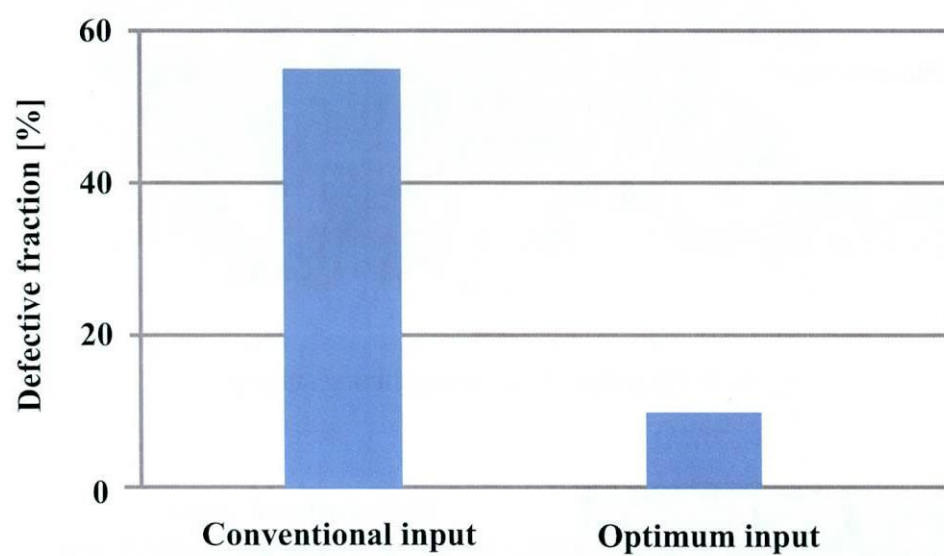


Fig. C.14 Defect fraction of machining surface

付 録 D 多極集中型最適化アルゴリズムの検証

D.1 初期探索点配置問題における遺伝的アルゴリズムを用いた時の収束性能

次に初期探索点に分散制御を適用したときの有効性の検証を行う。検証方法としては、探索点の生成を従来と同じくランダムで配置を行い、そこから分散制御則を用いて探索点同士を適当な間隔に配置する。後の処理は分散制御は用いず、遺伝的アルゴリズムによって生成する。このときのパラメータは Table 5.1 と同様の物を用いた。4 回実行して検証を行い、その結果をまとめた物を Table D.1 に示す。ここで、"Conv."は収束世代、"dist."は分散制御を意味する。Table D.1 を見てみると、4 つのパターンの内、パターン 3 の分散前を使用した場合には収束世代が 1 世代で終わり、また評価も周りと比べて 10 倍近く悪いことから局所解に陥っていることがわかる。対して分散後を利用した場合はほぼ一定の評価値を示していることから、良好に解析を行えていることがわかり、このことから分散前は分散後に比べ局所解に陥りやすいということがわか

Table D.1 Optimization results to compare the use of inithial population with the distribution and without

pattern 1	Cost	Conv.	(x [m/s], y [m])
Without dist.	1.37×10^{-3}	2	(0.25, 0.365)
With dist.	1.46×10^{-3}	3	(0.24, 0.345)

pattern 2	Cost	Conv.	(x [m/s], y [m])
Without dist.	1.46×10^{-3}	5	(0.18, 0.146)
With dist.	0.26×10^{-3}	11	(0.32, 0.339)

pattern 3	Cost	Conv.	(x [m/s], y [m])
Without dist.	9.90×10^{-3}	1	(0.39, 0.296)
With dist.	1.20×10^{-3}	8	(0.20, 0.174)

pattern 4	Cost	Conv.	(x [m/s], y [m])
Without dist.	0.10×10^{-3}	5	(0.20, 0.123)
With dist.	0.29×10^{-3}	6	(0.32, 0.338)

る。さらに、収束世代を比較してみると、分散前の状態では収束する世代が低く、比較的評価の悪い解に収束することから収束性に加えて、交叉も良好に行えていないことがわかる。パターン4を見てみると、分散前の評価が他のパターンに比べて非常に良いことがわかる。しかし、パターン2の分散前と比較してみると収束世代は同じである。このことからパターン4の結果は、ランダムによる初期配置が偶然、良好な解空間へ配置されたため得られた結果であると思われる。分散制御則を適用したパターン2とパターン4の結果を見てみると、評価と解がともにほぼ同じ値を示していることがわかる。比べて、分散制御則を適用していない場合は最適解が全て異なった結果を示している。このことから分散制御則を適用した場合は局所解に陥り難く、さらには最良の解が見つけ易くなると言える。

全パターンの結果をまとめて見ると、求まった最適解の評価の最大と最小を比較すると分散前の方が最適解の幅が大きいことから分散前では安定した解(評価)は得られないことがわかり、初期探索点に分散制御則を用いた方が、より良好な結果を安定して算出することがわかる。

解析した全ての探索点を図に起こし、探索点配置の様子を見る。Fig. D.1 ~ Fig. D.4はそれぞれ分散前でおこなったパターン1 ~ パターン4の時の探索点の配置であり、Fig. D.5 ~ Fig. D.8はそれぞれ分散後でおこなったパターン1 ~ パターン4の時の探索点の配置である。なお、赤の四角点は初期配置をしめす。

Fig. D.1 ~ Fig. D.4を見てみると、配置されていない領域における解析は全く行われていないことがわかる。特にパターン3の評価に見られるように、Fig. D.3においては、最適解がある領域は全く解析が行われていないことがわかり、局所解に陥ったことがわかる。また今回の解析でパターン4において $(v, x) = (0.20[\text{m/s}], 0.123[\text{m}])$ が最小解であると求められた。Fig. D.4を見てみると、その最適解近傍で解析を多く行っていることが良くわかる。しかしながら、そこに集中して解析を行っているために、他の領域に対しては殆ど探索を行っていないことがわかる。この現象はパターン3と同様に起こっており、このことからパターン4はランダム配置が偶然良好であったといえる。同じくパターン3から最適解周辺に初期探索点が配置されていることから、エリート解を中心に交叉が行われていることがわかる。

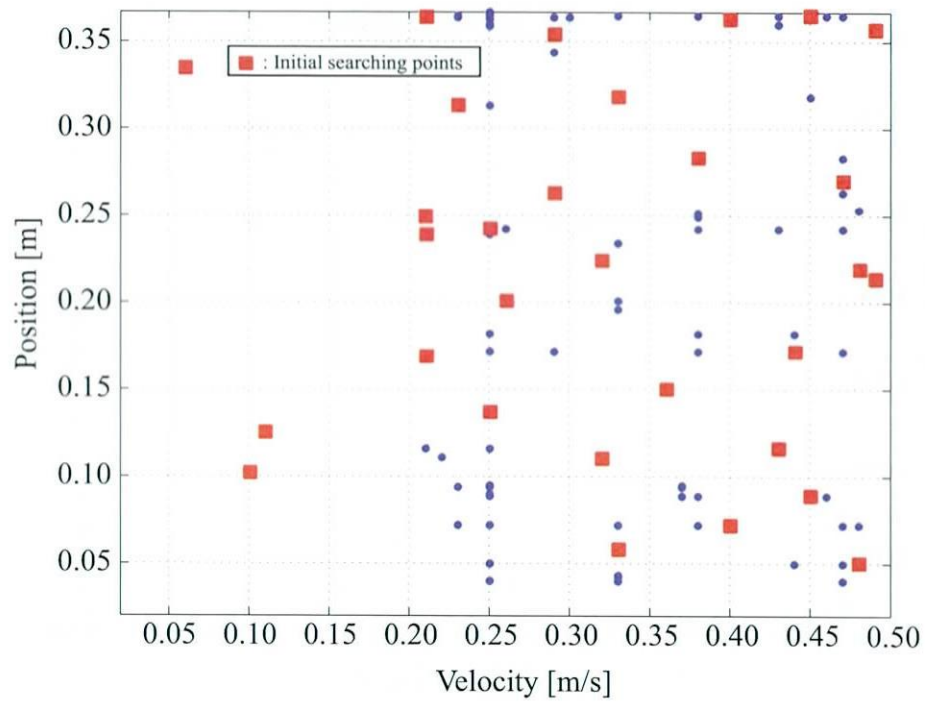


Fig. D.1 Location of all agents without distribute in pattern1

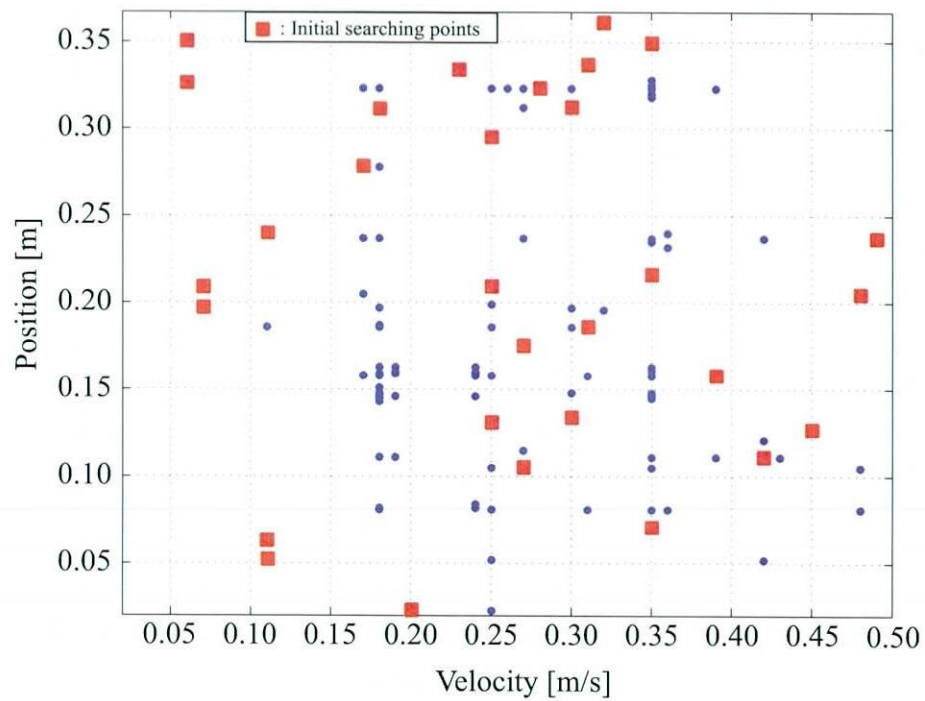


Fig. D.2 Location of all agents without distribute in pattern2

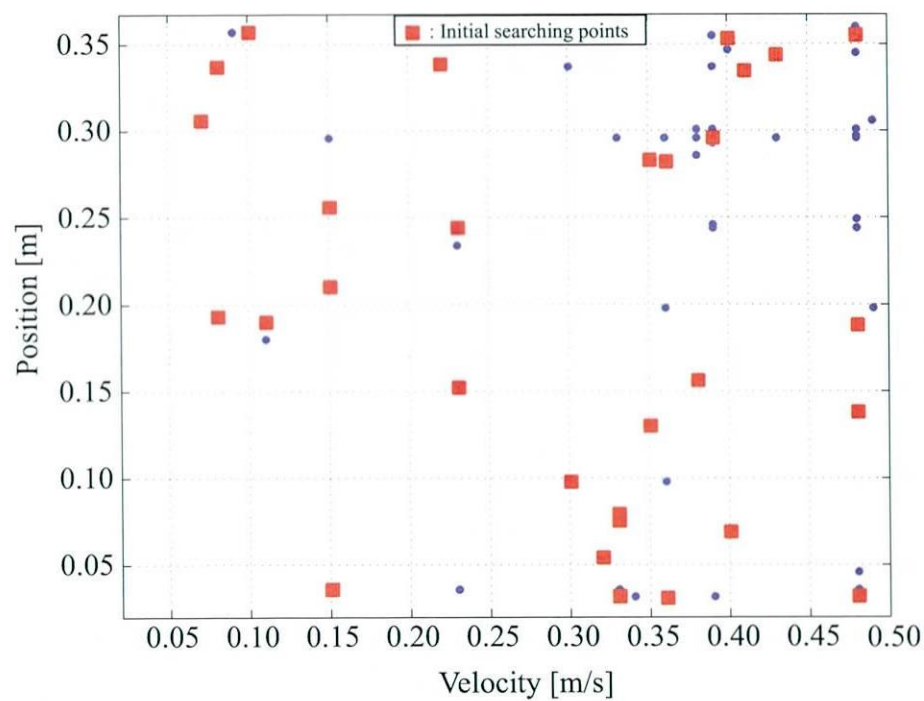


Fig. D.3 Location of all agents without distribute in pattern3

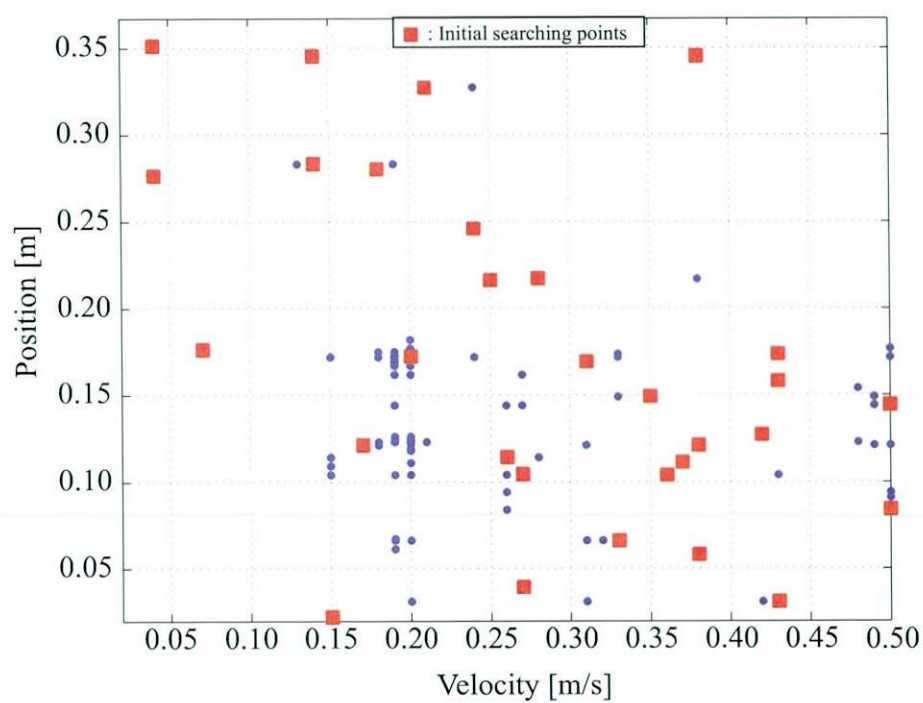


Fig. D.4 Location of all agents without distribute in pattern4

次に分散制御プログラムを適用した Fig. D.5 ~ Fig. D.8 を見てみる．初期配置を分散することで Fig. D.1 ~ Fig. D.4 に比べ広くに渡って解析をしていることが伺える．特に全体的に初期探索点以外が全体的に広がっていることから交叉も旨く行えていることがわかる．最良解はパターン 2 とパターン 4 であったが、いずれも最良解付近で多く探索点を配置していることがわかる．

最終的に遺伝的アルゴリズムで最適解を見つける場合、初期配置はランダムよりも分散制御則を用いて領域全体に探索点を配置される方が望ましいことがわかる．特に、遺伝的アルゴリズムの交叉性能の向上が Fig. D.5 ~ Fig. D.8 から確認できた．

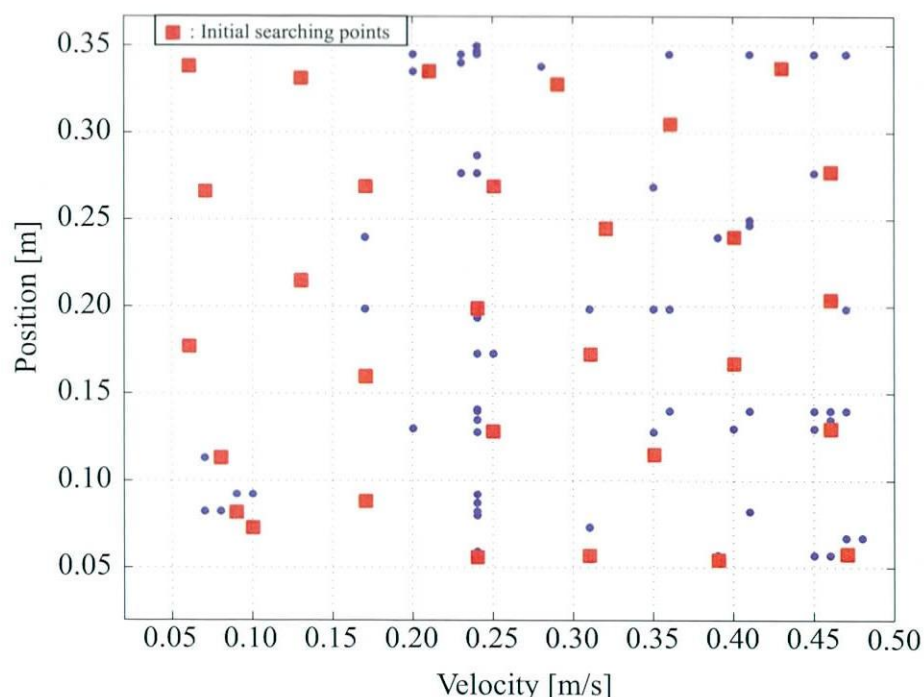


Fig. D.5 Location of all agents with distribute in pattern1

D.2 引力アルゴリズムを適用した場合の探索点の配置

Fig.D.9 と Fig.D.10 はそれぞれ引力アルゴリズムの設定をした場合と、していない場合を示している．ここで、パラメータの設定は Table 5.2 の数値を参考に、引力アルゴリズムの設定をそれぞれ $w_R = 0$, $b_a = 0$, $w_R = 0.1$, $b_a = 0.3$ とした．また Fig.D.9 と Fig.D.10 の探索点の数は同等である．Fig.D.9を見ると、探索点を多極に集中させていることがわかる．それに対して、引力アルゴリズムを設定した Fig.D.10を見ると、Fig.D.9 に比べ多極性を抑え、破線で示した一極に探索点が集中していることがわかる．これ

らの結果より，引力アルゴリズムの設定をすることで多極性が変化し，多極性を維持したい場合は引力アルゴリズムの設定を抑え，逆に大域的な最適解の一点を見つけた場合は引力アルゴリズムの設定を行う方が良いことがわかる．

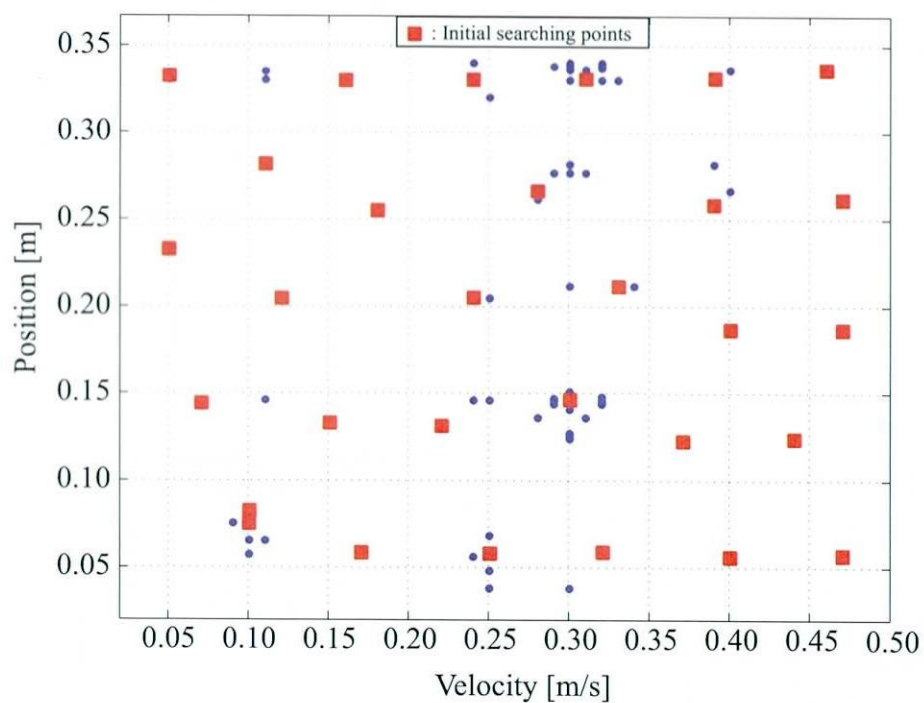


Fig. D.6 Location of all agents with distribute in pattern2

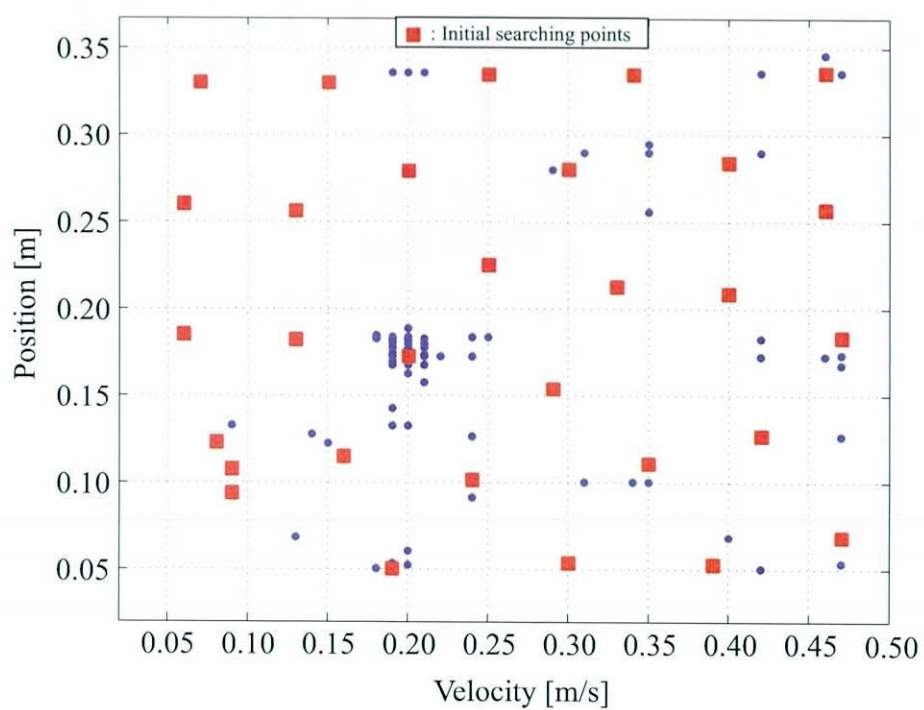


Fig. D.7 Location of all agents with distribute in pattern3

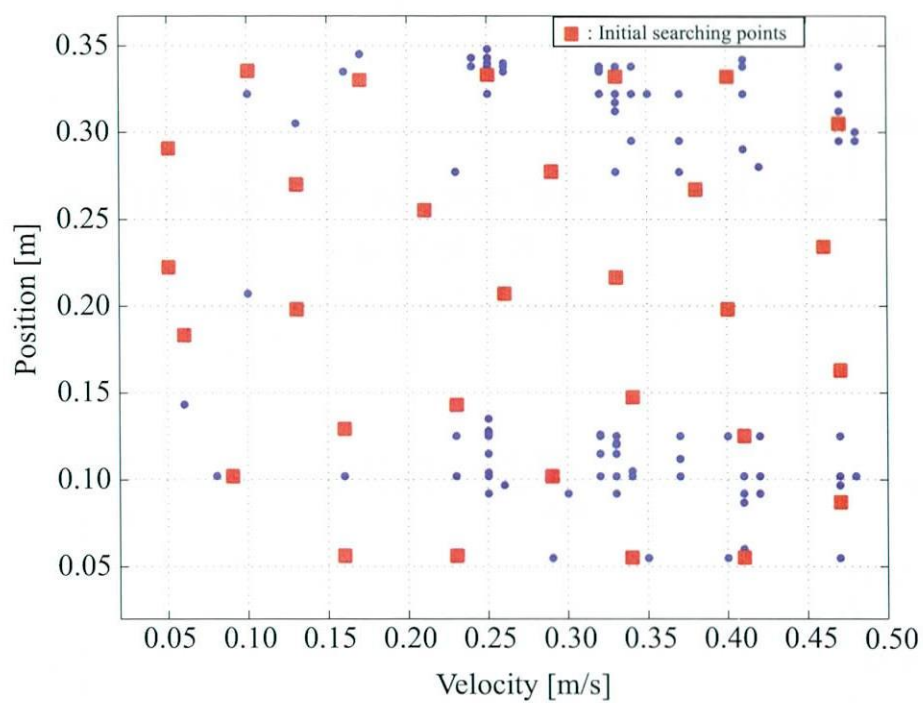


Fig. D.8 Location of all agents with distribute in pattern4

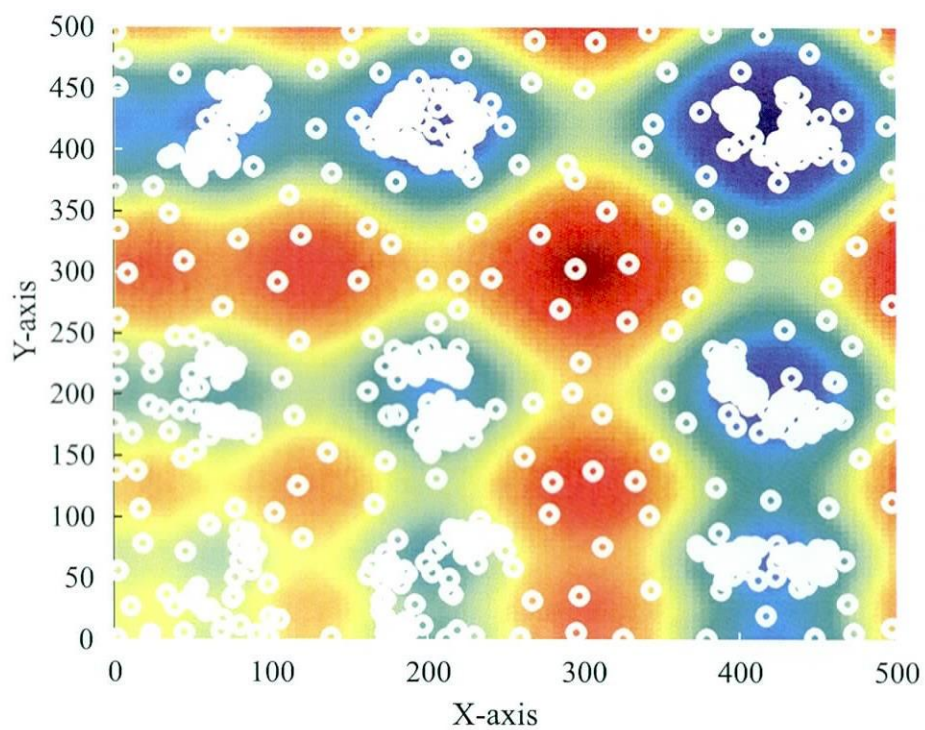


Fig. D.9 Location of the searching points using MSSA without attractive force algorithm

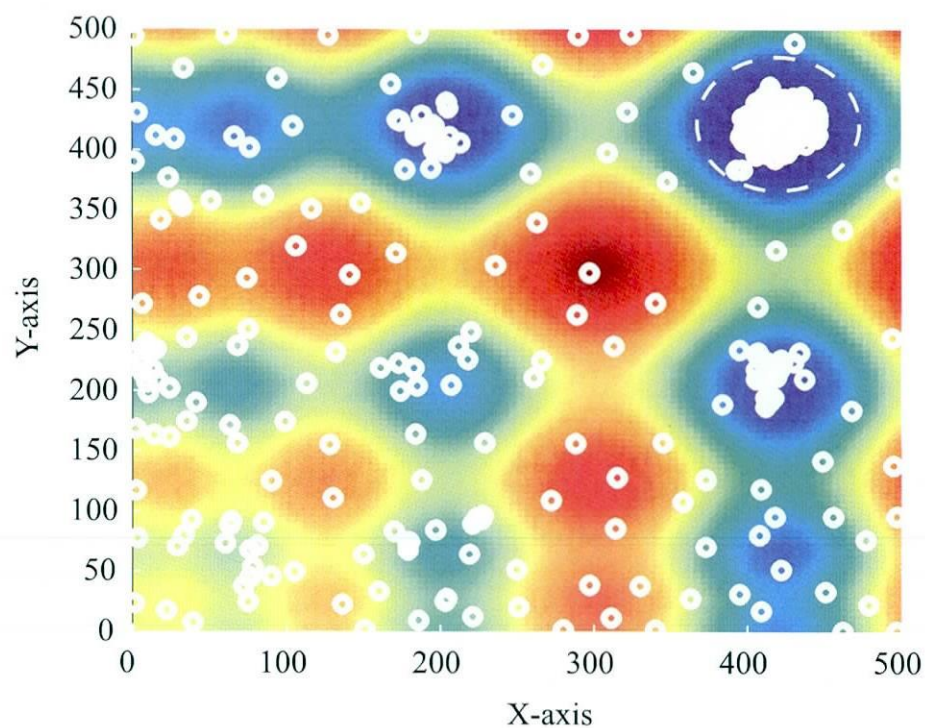


Fig. D.10 Location of the searching points using MSSA with attractive force algorithm

付録E 5変数における多極集中型最適化アルゴリズムの性能調査

E.1 ダイカストの射出速度最適化における収束性能の調査

平成23年1月までの解析済みの全データを用いて、ダイカストプランジャの最適速度入力の大域的な最適解を求め、これを利用して最適化性能の調査を行う。大域的な最適解の導出には、良好な評価を持つ解のデータから中央値で計算して求めた。ここで、中央値とはデータの分布中心を求めるもので、分布が対象でない場合に良く用いられる平均値に代わる計算法である。特に、測定ミスなどの外れ値の影響が小さく、また分布が対象の場合は平均値と同じ解を算出する。中央値はEq.(E.1)を用いて算出され、Eq.(E.1)を満たす中央値 m を求める。

$$\int_{-\infty}^m dF(x) \geq \frac{1}{2}, \quad \int_m^{\infty} dF(x) \geq \frac{1}{2} \quad (\text{E.1})$$

求めた結果、大域的な最適解は 0.41[m/s], 0.34[m/s], 0.57[m/s], 0.241[m], 0.322[m] であることがわかった。

次に、収束判定の定義を行う。本研究においてはシミュレータを用いて解析を行っているため、大域的な最適解発見までの到達世代で性能を比較することは非常に時間を要する。そこで大域的な最適解までの到達世代ではなく、大域的な最適解の近傍に到達した世代を収束世代として、収束性能の比較を行った。近傍の定義として、不偏分散の式を用いる。Eq.(E.2) は不偏分散の式を示しており、目標値からの分散度合い(ばらつき)を示すものである。

$$P = \frac{1}{n-1} \sum_n^{i-1} (x_i - \bar{x})^2 \quad (\text{E.2})$$

ここで、 P は評価値、 n は変数の数、 x_i は各軸の値、 \bar{x} はEq.(E.1)で求めた各軸の中央値を示す。このとき分散値を \sqrt{P} とし、 $\sqrt{P} \leq 1.5$ となったときを収束世代として算出、比較を行った。各試行毎の収束性能の結果をFig. E.1に示す。ここで、比較対象としては遺伝的アルゴリズムを用いて行い、各試行毎には同じ初期個体を用いて解析を開始している。また $\sqrt{P} \leq 1.5$ に到達しなかった場合は最大である60世代を収束世代とした。

解析の結果、遺伝的アルゴリズムに比べ、ばらつきが少なく、また収束世代も良好であることがわかる。収束世代の中央値を取ると、遺伝的アルゴリズムでは35世代だったのに対して、MSSAでは23.5世代で収束が行えていることがわかり、結果、33[%]の効率向上が見られた。なお、平均の場合は遺伝的アルゴリズムが34.7世代、MSSAが

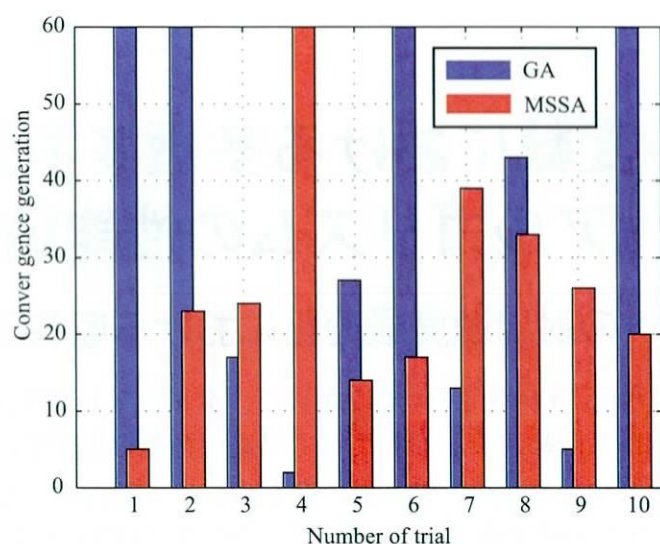


Fig. E.1 Convergence generation of the optimization trials

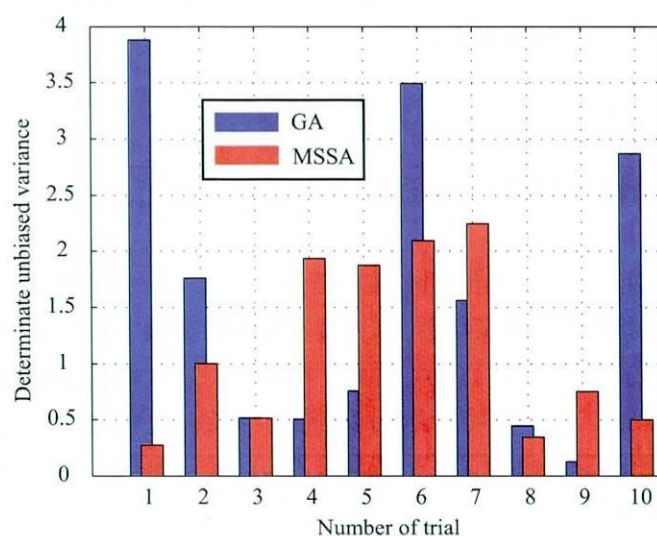


Fig. E.2 Determinate unbiased variance of the optimization trials

26.1 世代であり、25[%] の効率向上となる。また 60 世代まで計算して算出された解の分散度を調べた結果を Fig. E.2 に示す。これより、遺伝的アルゴリズムの試行 1 と試行 6 では非常に大域的な最適解から離れていることがわかる。分散度の中央値を取ると、遺伝的アルゴリズムは 1.1599、MSSA は 0.8706 であり、この結果から、MSSA では大域的な最適解へのアプローチが良好であることがわかった。なお、平均の場合は遺伝的アルゴリズムが 1.15929、MSSA が 1.1456 であった。

Fig. E.3 と Fig. E.4 に E.1 の収束時における最適値を示す。Fig. E.3 が遺伝的アルゴリズムによる結果、Fig. E.4 が MSSA による結果である。図中の破線で示す 5 角形が大域的な最適解を示す。これらの図からも、MSSA に比べ遺伝的アルゴリズムは局所解に陥ってしまっていることがわかる。

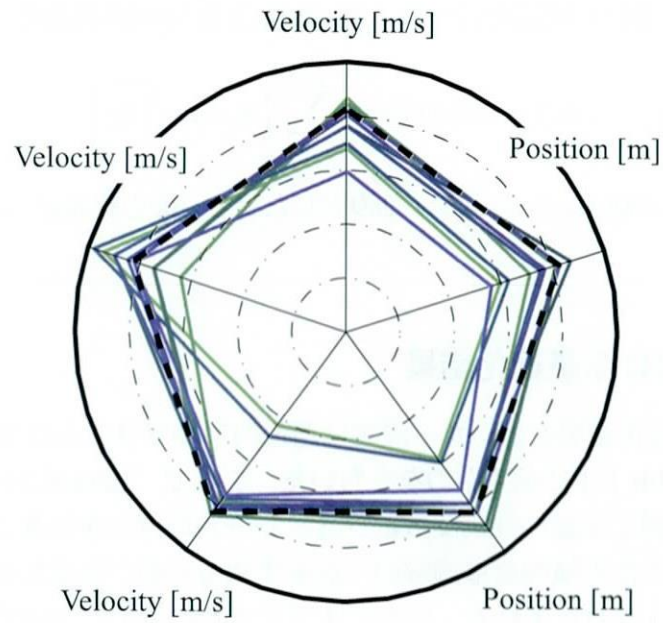


Fig. E.3 Optimum values of each trial by using GA

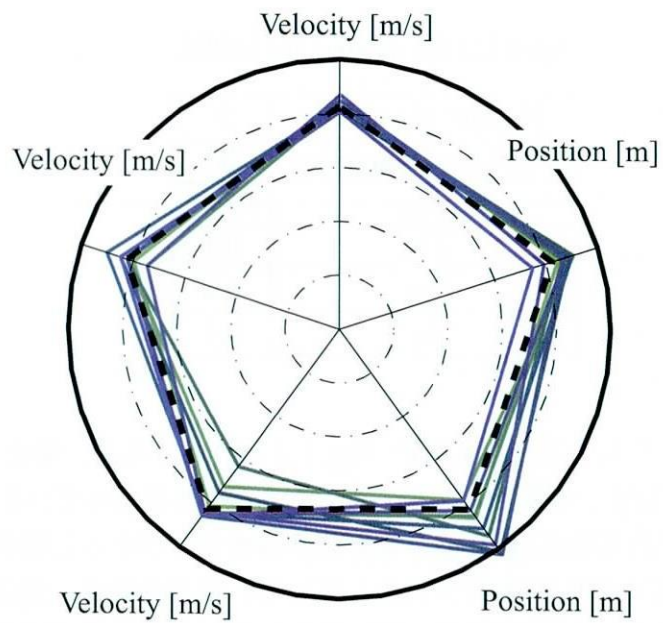


Fig. E.4 Optimum values of each trial by using MSSA

E.2 テスト関数を使った性能調査

E.2.1 Schwefel 関数

一般的な最適化問題で性能を比較するためテスト関数を使った性能調査を行う。テスト関数は多峰性解空間を持ち、かつ多変数への適用が可能なテスト関数のひとつで

ある Schwefel 関数を用いて検証を行った。Eq.(E.3) は Schwefel 関数の基礎式である。

$$f(x_1, \dots, x_n) = 418.9829n - \sum_{j=1}^n \left(x_j \sin \sqrt{|x_j|} \right) \quad (\text{E.3})$$

解析範囲を $0 \leq x_j \leq 500 (j = 1 \sim n)$ で 1 刻みとしたとき、最小解は $\mathbf{x}^* = [421, \dots, 421]$ となる。

E.2.2 2 変数における最適化結果

2 変数における最適化結果を行った。設定は遺伝的アルゴリズムは Table 5.1 と同じものを、MSSA は Table E.1 に示すものをもちいた。ここで、Table E.1 の設定については、ダイカストの問題とは違い評価関数にシミュレータの評価を加えていないため解空間が安定となり、また大域的な最適解の一点を求める必要があるため、引力アルゴリズムの設定 (w_R, b_a) を行っている。引力アルゴリズムの設定については、10 世代で R_{\max} の範囲を引力にするために $w_R = 0.1$ とし、また平均から 30[%] 以下の評価を引力中心とするため $b_a = 0.3$ とした。

Table E.1 Parameters for MSSA

Number of variable	2
Number of population	30
R_{add}	0.001
F_{lim}	0.1
w_R	0.1
b_a	0.3

Fig. E.6 は解析した解の配置状況を示す。Fig. E.5 と比較すると、座標 (421, 230) と座標 (230, 421) において探索点が集中していることから多極性をもっていることがわかる。

Fig. E.7 に最適解である座標 (421, 421) を発見するまでに掛かった収束世代をまとめたものを示す。なお最大世代数は 45 世代としている。解析の結果、遺伝的アルゴリズムでは 7 回目の試行の時に局所解に陥っていることがわかる。収束世代の中央値を取ると、遺伝的アルゴリズムでは 24 世代だったのに対して、MSSA では 14 世代で収束が行えていることがわかり、結果、42[%] の探索性能の効率向上が見られた。なお、平均の場合は遺伝的アルゴリズムが 27 世代、MSSA が 15.2 世代であり、44[%] の効率向上となる。

不偏分散度で収束性能を評価した場合を Fig. E.8 に示す。このとき $\sqrt{P} \leq 0.5$ となったときを収束世代として算出、比較を行う。また $\sqrt{P} \leq 0.5$ に到達しなかった場合は最大である 45 世代を収束世代とした。

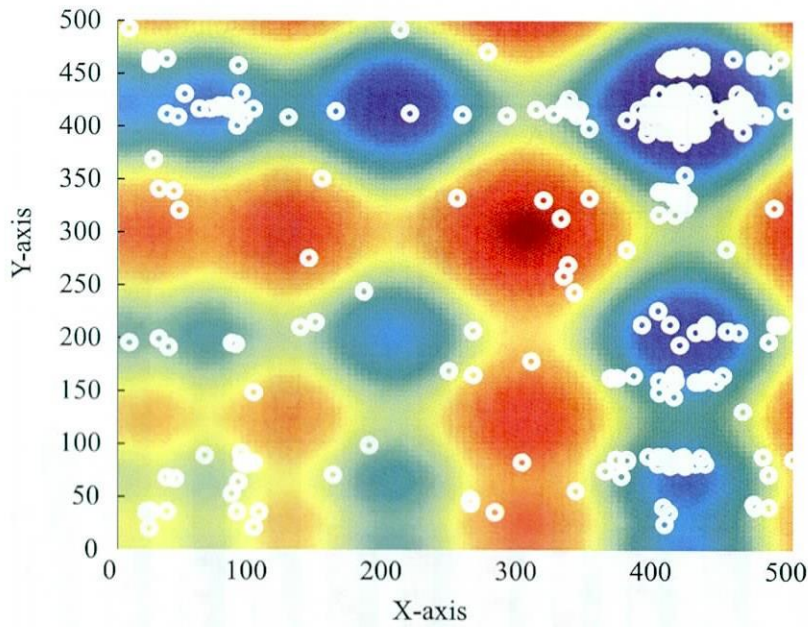


Fig. E.5 Location of the searching points using GA

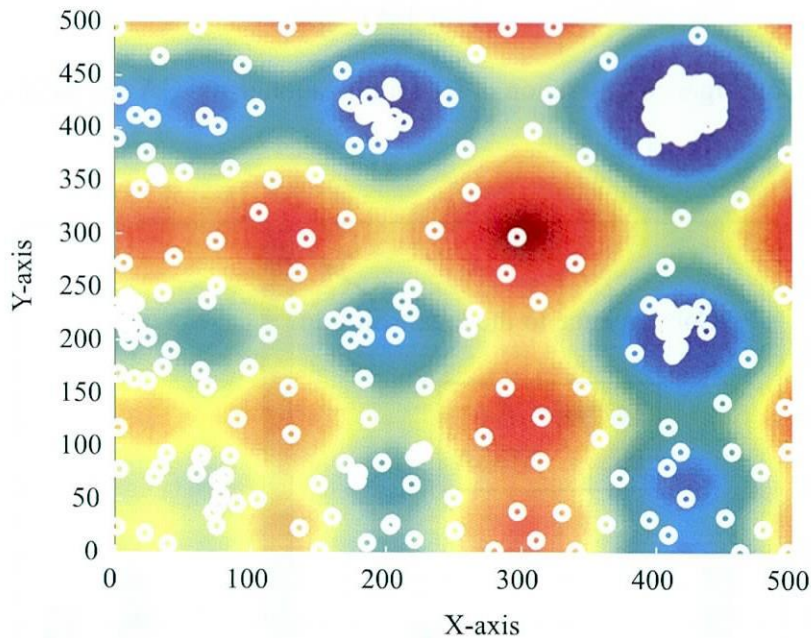


Fig. E.6 Location of the searching points using MSSA

解析の結果、遺伝的アルゴリズムに比べ、収束世代が良好であることから、大域的な最適解近傍への到達率が早いことがわかる。収束世代の中央値を取ると、遺伝的アルゴリズムでは14世代だったのに対して、MSSAでは7世代で収束が行えていることがわかり、結果、50[%]の効率向上が見られた。なお、平均の場合は遺伝的アルゴリズムが16.5世代、MSSAが6.7世代であり、59[%]の効率向上となる。

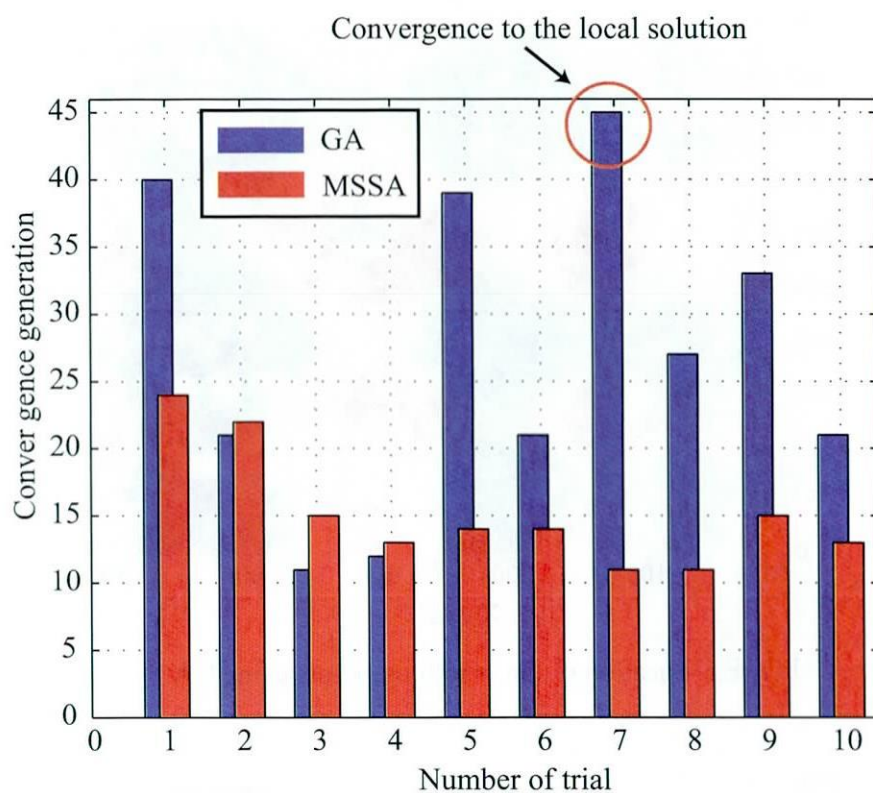


Fig. E.7 Convergence generation of the optimization using GA and MSSA

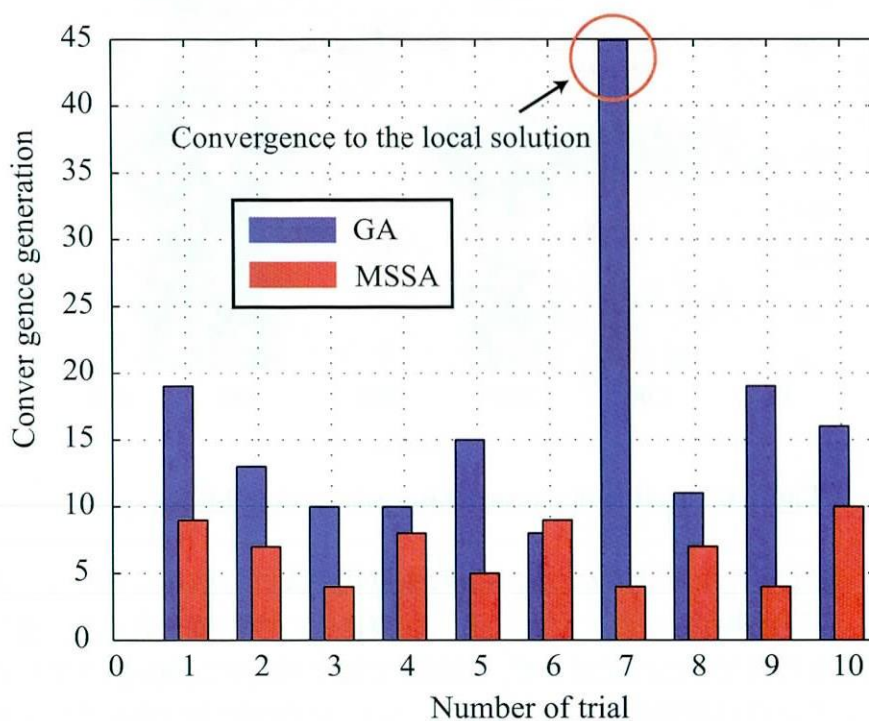


Fig. E.8 Convergence generation with median of the optimization using GA and MSSA

E.2.3 5変数における最適化結果

最後に5変数の最適化結果を示す。最適化は5回の試行を行い、最適解である座標(421, 421, 421, 421, 421)を発見するまでに掛かった収束世代を算出する。Fig. E.9に収束世代をまとめたものを示す。なお最大世代数は250世代としている。収束世代の中央値を取ると、遺伝的アルゴリズムでは143世代だったのに対して、MSSAでは64世代で収束が行えていることがわかり、結果、55[%]の探索性能の効率向上が見られた。なお、平均の場合は遺伝的アルゴリズムが171世代、MSSAが72.2世代であり、48[%]の効率向上となる。

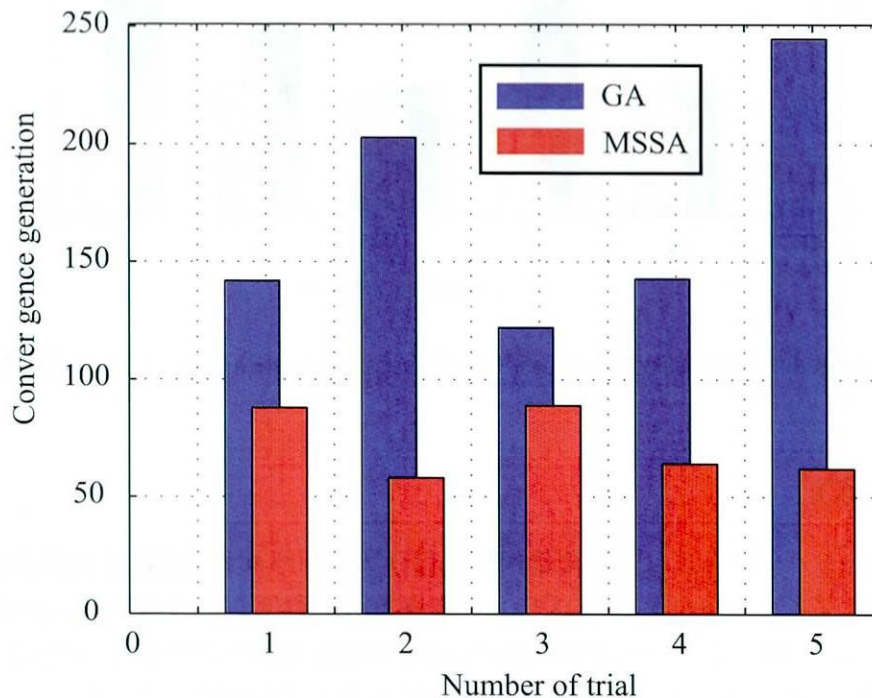


Fig. E.9 Convergence generation of the optimization using GA and MSSA (5 variables)

不偏分散度で収束性能を評価した場合をFig. E.10に示す。このとき $\sqrt{P} \leq 0.5$ となったときを収束世代として算出、比較を行う。

解析の結果、遺伝的アルゴリズムに比べ、収束世代が良好であることから、大域的な最適解近傍への到達率が早いことがわかる。収束世代の中央値を取ると、遺伝的アルゴリズムでは48世代だったのに対して、MSSAでは29世代で収束が行えていることがわかり、結果、40[%]の効率向上が見られた。なお、平均の場合は遺伝的アルゴリズムが62.4世代、MSSAが28.6世代であり、54[%]の効率向上となる。

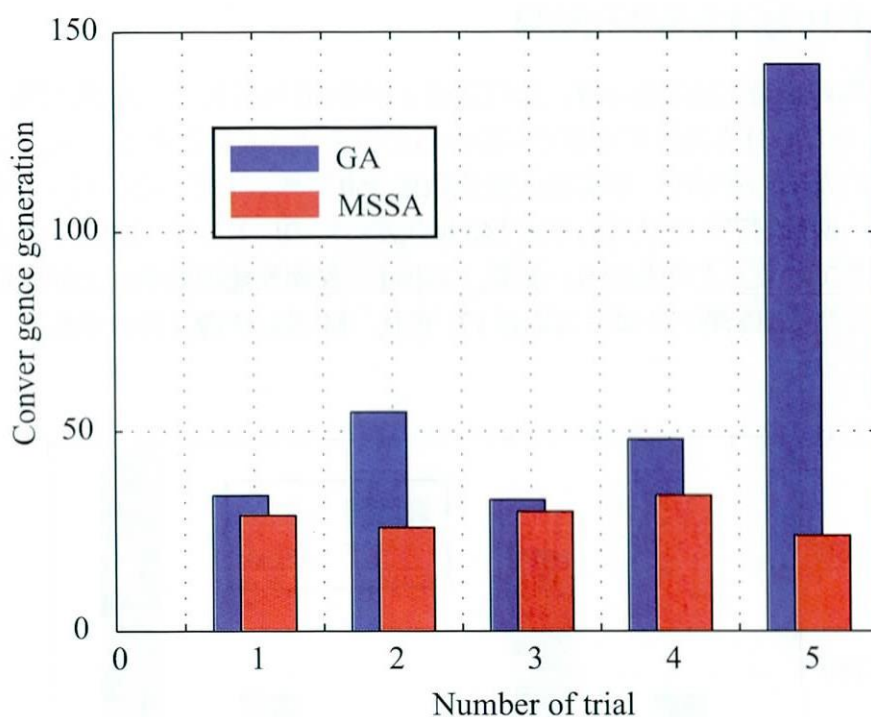


Fig. E.10 Convergence generation with median of the optimization using GA and MSSA (5 variables)

E.3 まとめ

本章では遺伝的アルゴリズムと多極集中型最適化アルゴリズム (MSSA) の検証を行った。具体的にはダイカストにおけるプランジャ速度入力の大域的な最適解への収束性能の比較と、多峰性解空間の一つである Schwefel 関数を用いて収束性能の比較を行った。結果、遺伝的アルゴリズムに比べ多極集中型最適化アルゴリズムは収束性能の向上がみられ、多極集中型最適化アルゴリズムの有効性を示すことができた。

極値近傍探索アルゴリズムにおいては、大域的な最適解への収束が難しく、全ての試行において大域的な最適解を見つけることが確認された。これは近似曲線の生成に失敗していることが原因と考えられる。特に Schwefel 関数で大域的な最適解の導出が難しいことから多様性に欠けると言える。

付録F ダイカストプランジャの最適速度入力に関する追加実験

これまで提案した最適化アルゴリズムの検証を行った。実験は低速射出条件 (Low Speed), 高速射出条件 (High Speed), 遺伝的アルゴリズムで求めた解 (GA), 極値近傍探索アルゴリズムで求めた解 (EDSA), 多極集中アルゴリズムで求めた解 (MSSA), 多極集中アルゴリズムに空間探索アルゴリズムを適用して求めた解 (MSSA+SSA) の合計6条件をもちいて行った。このとき使用した各パラメータと射出完了時間を Table F に示す, 実験結果を Fig. F.1 ~ Fig. F.6 に示し, 各射出条件におけるブリストー試験後の気泡面積についてまとめたものを Fig. F.7 に示す。Fig. F.7 から, 各最適化アルゴリズムは低速および高速射出条件 (Low Speed & High Speed) に比べ気泡面積が少なくなっていることがわかる。実験でも良好な解を得られることを目的とした最適化アルゴリズムである極値近傍探索アルゴリズム (EDSA) と, 多極集中アルゴリズムに空間探索アルゴリズムを組み込んだアルゴリズム (MSSA+SSA) による結果では, 全体的に良好な結果を得ていることがわかる。ついで収束世代が良好であった多極集中アルゴリズム (MSSA) では遺伝的アルゴリズム (GA) と同様に良好な結果を示している。また多極集中アルゴリズム (MSSA) は極値近傍探索アルゴリズム (EDSA) に比べ 150 ~ 300 回ほど繰り返し計算回数の低減が見込まれるとともに, MSSA は前章 (付録 E) にて Schwefel 関数でも有効性を示すことができた。したがって多極集中アルゴリズムに空間探索アルゴリズムを組み込んだアルゴリズム (MSSA+SSA) を用いることで, すばやくかつ良好な解を得ることができる。

Table F.1 Plunger velocity parameter of each input.

	x_1 [m]	x_2 [m]	v_1 [m/s]	v_2 [m/s]	v_3 [m/s]	t_{fin} [s]
Low Speed	0.05	0.1	0.21	0.21	0.21	2.86
High Speed	0.05	0.1	0.50	0.50	0.50	1.20
GA	0.182	0.252	0.35	0.4	0.33	1.54
EDSA	0.075	0.105	0.27	0.24	0.46	1.42
MSSA	0.261	0.355	0.42	0.33	0.48	1.52
MSSA+SSA	0.274	0.353	0.41	0.34	0.46	1.58

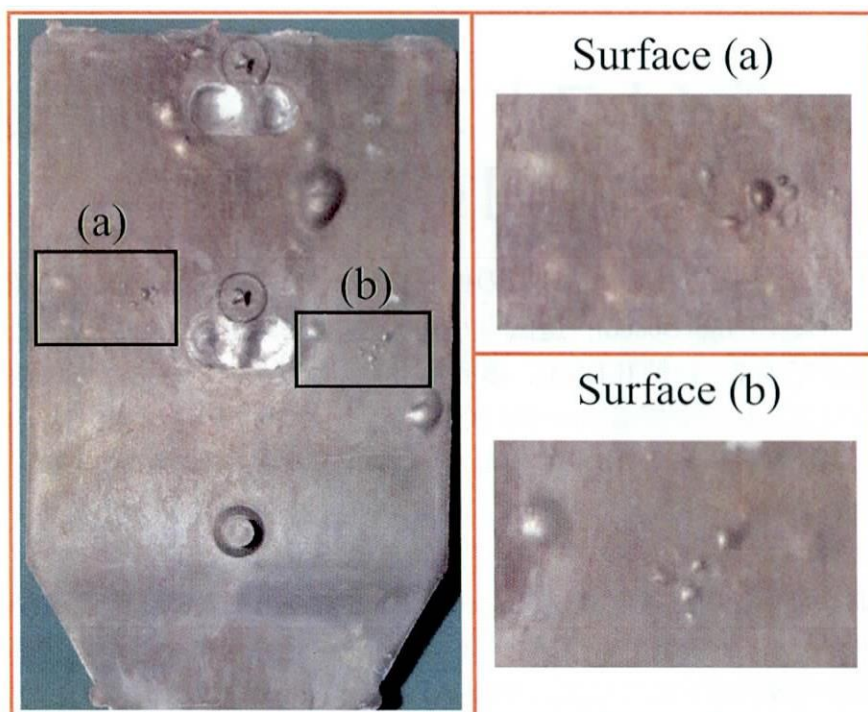


Fig. F.1 Experimental results of low speed injection

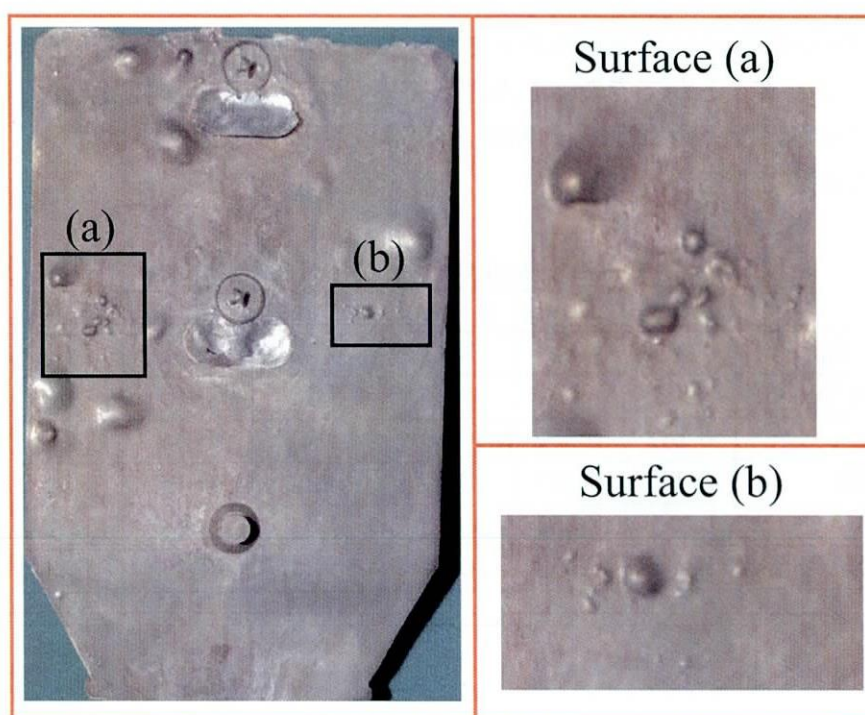


Fig. F.2 Experimental results of high speed injection

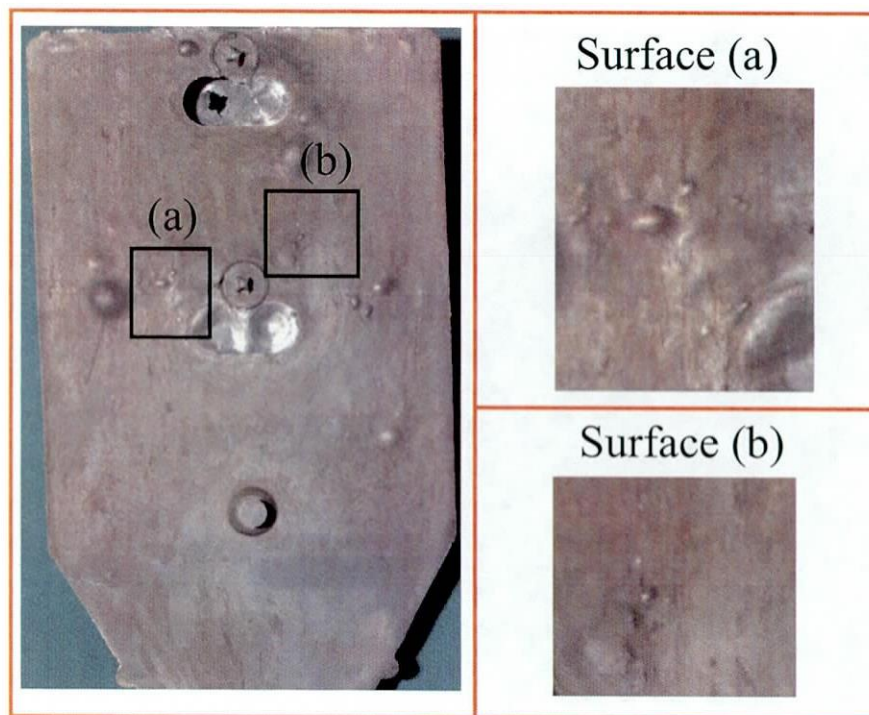


Fig. F.3 Experimental results of the optimum injection derived by GA

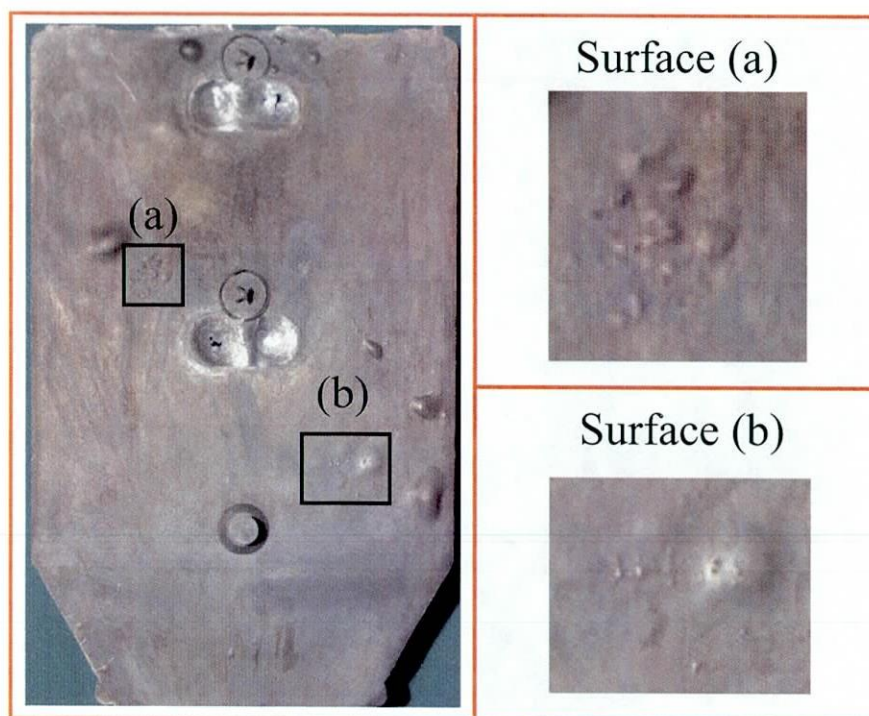


Fig. F.4 Experimental results of the optimum injection derived by EDSA

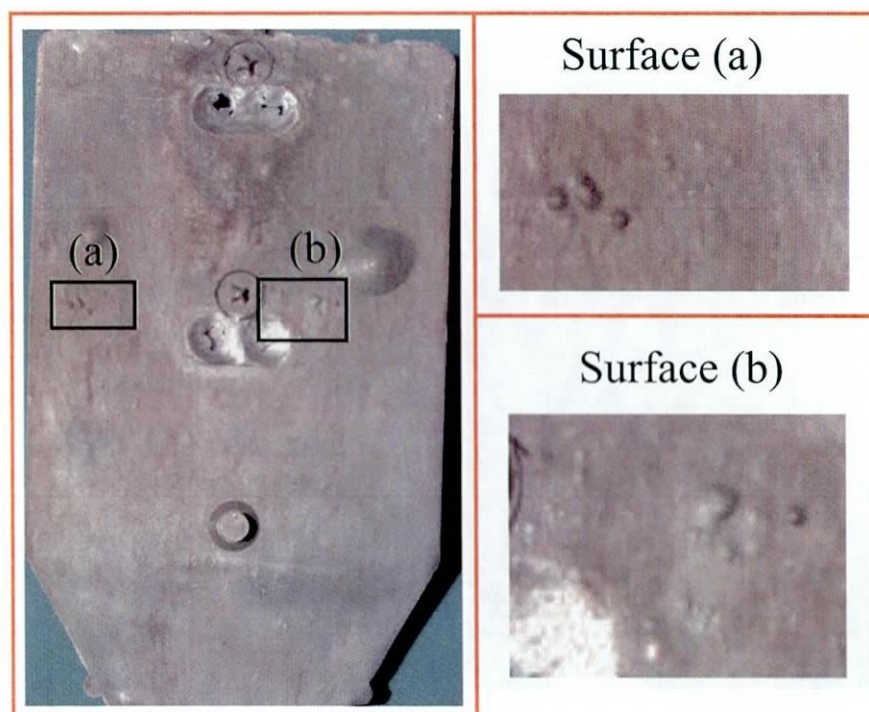


Fig. F.5 Experimental results of the optimum injection derived by MSSA

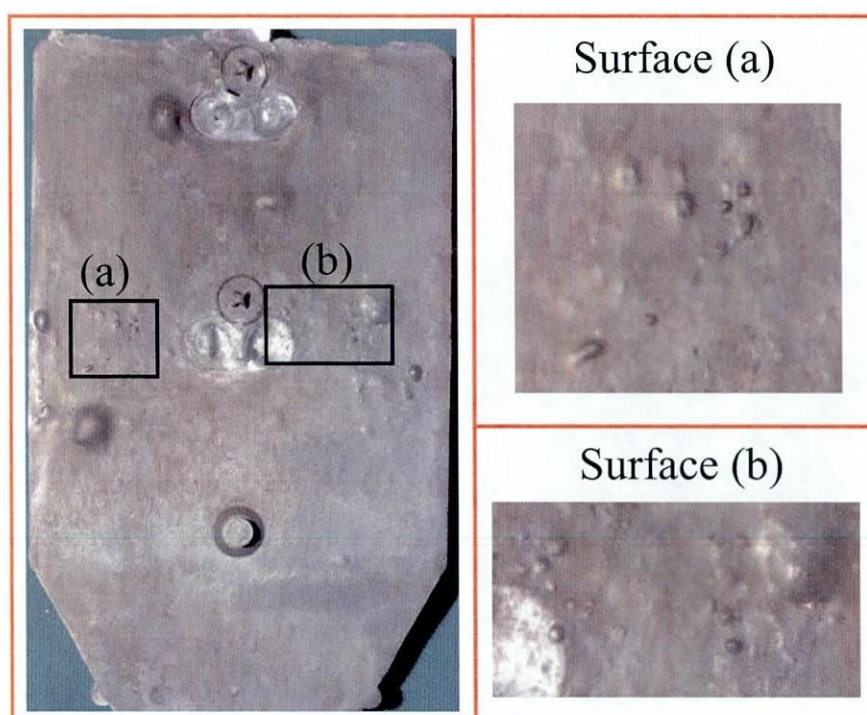


Fig. F.6 Experimental results of the optimum injection derived by MSSA with SSA

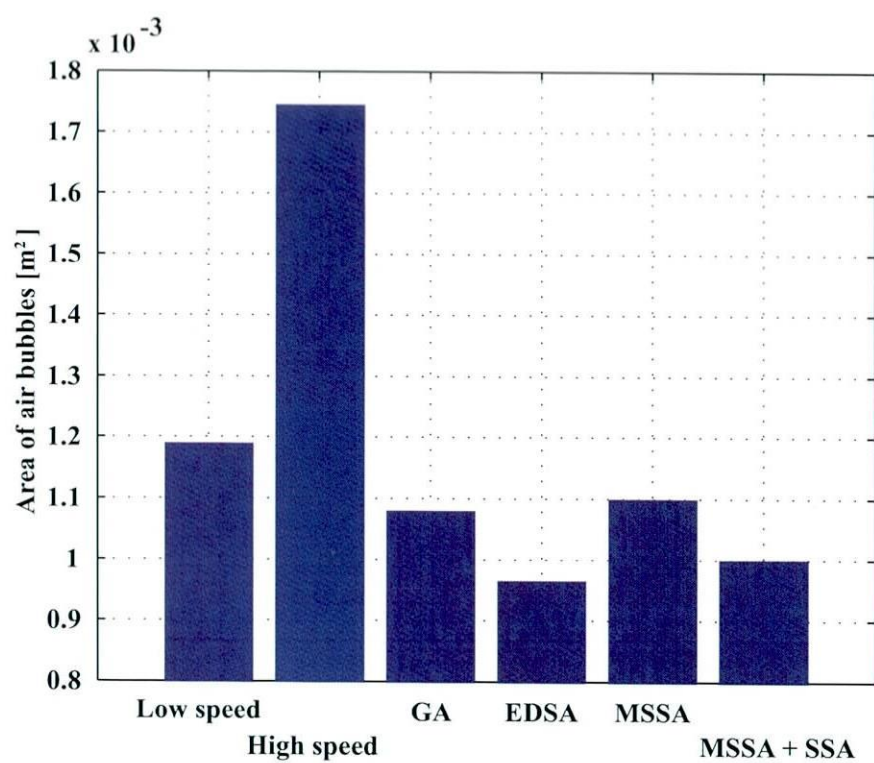


Fig. F.7 Area of air bubbles of each condition

目 次

1.1	Comparison of the fluid behavior between experimental result and simulation result	2
1.2	A conception diagram of die casting and CFD simulation	4
1.3	Multimodal search space	8
1.4	Cost function of die casting process	8
2.1	Picture of 6DOF manipulator	12
2.2	Details of the spoon shape	13
2.3	Mesh setting of the spoon	14
2.4	Reference and experimental velocity of the spoon	15
2.5	Experimental result of the spoon transfer	16
2.6	Simulation result of the spoon transfer	16
2.7	Setup coordinate	17
2.8	Experimental result of angle	21
2.9	Experimental result of angle	21
2.10	Experimental result of position	22
2.11	Experimental result of orientation	22
2.12	Experimental result of position	23
2.13	Identify 1st motor of step input	25
2.14	Block diagram of feedback for motor control	25
2.15	Identify 1st motor of sine wave input	26
2.16	Identify 2nd motor of step input	27
2.17	Identify 2nd motor of sine wave input	28
2.18	Identify 3rd motor of step input	28
2.19	Identify 3rd motor of sine wave input	29
2.20	Identify 4th motor of step input	30
2.21	Identify 4th motor of sine wave input	31
2.22	Identify 5th motor of step input	31
2.23	Identify 5th motor of sine wave input	32
2.24	Identify 6th motor of step input	33
2.25	Identify 6th motor of sine wave input	34
2.26	Direction of the spoon transfer	35
2.27	Measurement plane of the spoon	36

2.28	Planning of velocity curve to derive the transfer path with spilling avoidance	37
2.29	Conceptual diagram of Genetic Algorithm	39
2.30	Example of one-point crossover	39
2.31	Optimized result of velocity and acceleration curves	40
2.32	Angular velocity of 6DOF manipulator	41
2.33	Fluid analysis simulation result of liquid transfer	41
2.34	Experimental results of each angler velocity of 6DOF manipulator by the proportional control	44
2.35	Experimental results of each angler velocity of 6DOF manipulator by the hybrid shape approach	44
2.36	Experimental results of each angler velocity of 6DOF manipulator by the proposed method	45
2.37	Transfer path by the each controller	45
2.38	Experimental result by the proportional control	46
2.39	Experimental result by the hybrid shape approach	47
2.40	Experimental result by the proposed method	48
3.1	Picture of the cold chamber die casting machine and the mold	53
3.2	Results of a brister test ($v_\ell = 0.5[\text{m/s}]$, $v_h = 1.0, 2.0[\text{m/s}]$)	53
3.3	Entrainment model	55
3.4	Mesh setting for CFD simulation	56
3.5	Result of air entrainment searched all velocity	58
3.6	Simulation result of $v_\ell = 0.50[\text{m/s}]$	59
3.7	Simulation result of $v_\ell = 0.21[\text{m/s}]$	59
3.8	Result of a brister test ($v_\ell = 0.21[\text{m}]$, $v_h = 2.0[\text{m/s}]$)	60
3.9	Simulation result of $v_\ell = 0.26[\text{m/s}]$	60
3.10	Result of a brister test ($v_\ell = 0.26[\text{m}]$, $v_h = 2.0[\text{m/s}]$)	61
3.11	Planger velocities of each result	61
3.12	Distinction of the shutting the air in the sleeve	62
3.13	Velocity curve of a plunger	63
3.14	Optimization result	65
3.15	Velocity curve of optimum velocity with 5 variables	66
3.16	CFD simulation result using optimum velocity with 5 variables	66
3.17	Result of a brister test using optimum velocity with 5 variables	68
3.18	Result of a brister test using optimum velocity with 2 variables	68
4.1	Concept of the EDSA	70
4.2	Extremum plot example	71
4.3	Individual selection	73

4.4	Simplex crossover	74
4.5	Flow chart of the EDSA	76
4.6	The cost function of the GA and EDSA	77
4.7	Optimum plunger velocity input of GA	79
4.8	Optimum plunger velocity input of EDSA	79
4.9	Optimum plunger velocity of GA	80
4.10	Optimum plunger velocity of EDSA	80
4.11	Another result of cost function of GA and EDSA	81
4.12	Optimum plunger velocity input of GA	81
4.13	Optimum plunger velocity input of EDSA	82
4.14	Optimum plunger velocity of GA	82
4.15	Optimum plunger velocity of EDSA	83
4.16	Result of brister test of GA	86
4.17	Result of brister test of EDSA	87
4.18	Another result of brister test for GA	88
4.19	Another result of brister test for EDSA	89
5.1	Flowchart of the Multi-subcenters Solution Search Algorithm	92
5.2	Generated agents by using random	93
5.3	Basic concept of distribute control	94
5.4	Setting the searching point of next generation	96
5.5	Moving concept of searching point	96
5.6	Attractive force algorithm	98
5.7	Arranged agents by using distribute control after the random location	99
5.8	Example of relationship between two variables and air entrainment	100
5.9	Convergence performance comparison between MSSA and GA	101
5.10	Optimum velocity curve calculated by GA	102
5.11	Optimum velocity curve calculated by MSSA	102
5.12	Simulation result by using optimum velocity curve calculated by GA	103
5.13	Simulation result by using optimum velocity calculated by MSSA	103
5.14	Experimental result by using optimum velocity calculated by GA	104
5.15	Experimental result of optimization calculated by MSSA	105
6.1	Location of search points by using multi-subcenters solution search algorithm	108
6.2	Multi-subcenters Solution Search Algorithm adoption of Space Searching Algorithm	110
6.3	Velocity curve by using the MSSA result modified by SSA	111
6.4	Behavior of the fluid by using the MSSA result modified by SSA	111
6.5	Experimental result by using the MSSA result modified by SSA	113

C.1	Overview of the automatic pouring machine	138
C.2	Tilting velocity of pouring machine	139
C.3	Result of identification	140
C.4	Mesh setting of pouring machine	141
C.5	Tilting input for the parameter identification	142
C.6	Comparison of flow line between experiment and CFD simulation	143
C.7	Comparison of flux in sprue cup between experiment and CFD simulation	144
C.8	Cost function	145
C.9	Tilting input of optimization result	146
C.10	Comparison of surface height between optimal input and conventional input	147
C.11	Simulation result with optimum tilting velocity	148
C.12	Air entrainment at machining surface	149
C.13	Experimental results	149
C.14	Defect fraction of machining surface	150
D.1	Location of all agents without distribute in pattern1	153
D.2	Location of all agents without distribute in pattern2	153
D.3	Location of all agents without distribute in pattern3	154
D.4	Location of all agents without distribute in pattern4	154
D.5	Location of all agents with distribute in pattern1	155
D.6	Location of all agents with distribute in pattern2	157
D.7	Location of all agents with distribute in pattern3	157
D.8	Location of all agents with distribute in pattern4	158
D.9	Location of the searching points using MSSA without attractive force algorithm	159
D.10	Location of the searching points using MSSA with attractive force algorithm	159
E.1	Convergence generation of the optimization trials	162
E.2	Determinate unbiased variance of the optimization trials	162
E.3	Optimum values of each trial by using GA	163
E.4	Optimum values of each trial by using MSSA	163
E.5	Location of the searching points using GA	165
E.6	Location of the searching points using MSSA	165
E.7	Convergence generation of the optimization using GA and MSSA	166
E.8	Convergence generation with median of the optimization using GA and MSSA	166
E.9	Convergence generation of the optimization using GA and MSSA (5 variables)	167
E.10	Convergence generation with median of the optimization using GA and MSSA (5 variables)	168

F.1	Experimental results of low speed injection	170
F.2	Experimental results of high speed injection	170
F.3	Experimental results of the optimum injection derived by GA	171
F.4	Experimental results of the optimum injection derived by EDSA	171
F.5	Experimental results of the optimum injection derived by MSSA	172
F.6	Experimental results of the optimum injection derived by MSSA with SSA .	172
F.7	Area of air bubbles of each condition	173

表 目 次

2.1	Maximum angular velocity of the motors	12
2.2	Setting mesh parameters of the spoon	13
2.3	Fluid parameters of water	14
2.4	Link parameter of 6DOF manipulator	18
2.5	Parametar of each motion motor	33
2.6	Parameters for genetic algorithm	40
3.1	Physicality value (Fluid properties ADC12)	54
3.2	Physicality value (Mold properties SKD61)	54
3.3	Mesh Parameters for die-casting	57
3.4	Quantity of air entrainment	58
3.5	Weight of evaluation value	64
3.6	Parameters for Genetic algorithm	64
3.7	Optimum plunger velocity	66
3.8	Optimum plunger velocity	67
4.1	Optimization results of the GA and EDSA	77
4.2	Another optimization results of GA and EDSA	78
4.3	Plunger velocity parameters	85
4.4	Plunger velocity parameters of the another optimization results	85
5.1	Parameters for GA	99
5.2	Parameters for MSSA	100
5.3	Performance comparison of optimization results	102
5.4	Experimental first result of optimization calculated by GA	104
5.5	Experimental result by using optimum velocity calculated by MSSA	105
6.1	Classification of multivariate statistical methods	108
6.2	Simulation result using the MSSA result modified by SSA	109
6.3	Optimum velocity modified by SSA	112
C.1	Setting of the tilting input	139
C.2	Fluid paramerets of AC2B	140
C.3	Mesh parameters of pouring machine	141
C.4	Input setting for the parameter identification	141

C.5	Parameters of porous baffle	142
C.6	Parameters for GA	145
C.7	Input setting of optimization result	146
C.8	Simulation results of air entrainment	147
D.1	Optimization results to compare the use of initial population with the distribution and without	151
E.1	Parameters for MSSA	164
F.1	Plunger velocity parameter of each input.	169