

DSM とシミュレーションを用いた開発プロセスの提案

Proposal of Development Process Using DSM and Simulation

藤 本 茂 希*

Shigeki FUJIMOTO

楠 祥 治*

Shoji KUSUNOKI

東 上 圭 司*

Keiji TOJO

齊 藤 徹*

Toru SAITO

要 旨

開発の初期段階からシステムの目標性能やそれを達成する手段を定めることで、手戻りの無い要求型開発をおこなうため、開発プロセスを構築した。このプロセスでは Design Structure Matrix を用いて合理的な開発順序を決め、シミュレーションによりシステムの目標性能達成可否を予測すると共に、目標性能が未達の場合に備え他の案別を網羅的に計算する。その結果にデータマイニング技術を用いることで、複数の解候補から素早く合理的な解を抽出する。事例としてパワートレインシステム開発を題材に、燃費性能、発進加速、低回転こもり音領域の性能検討をおこない、システム設計者達が最適なバランス解に絞り込めることを検証した。

ABSTRACT

A development process was created to perform request-type development without reworking by determining the target system performance and the means for achieving that performance from the initial development stage. This process uses a Design Structure Matrix to determine the most reasonable development order, performs simulations to predict whether the target system performance can be achieved, and comprehensively calculates alternate solutions in case the target performance is not achieved. Data mining technology is then applied to these results to quickly extract the most reasonable solution from among multiple solution candidates. By way of illustration, using power train system development as an example, performance was investigated for the areas of fuel economy performance, acceleration from standstill, and low-engine-speed booming noise, and it was verified that the system designers could narrow down the optimal balance solution.

1. まえがき

近年の車両開発はグローバル化による仕向地拡大に伴い、パワートレインの多様化が進んでいる。また、安全ニーズの高まりから予防安全システムが搭載され、デバイスや制御の複雑化も加速している。一方、開発期間やコストには限りがあるため、このような多様で複雑な自動車システムを効率的に開発する技術が求められている。

V-model は、開発するシステムの設計工程とテスト工程を、その粒度に合わせて階層化し検証レベルを一致させる考え方である⁽¹⁾。またモデルベース開発(MBD)は、設計の初期工程からモデルを活用し、テスト工程でおこなうべき検証の前倒しを図る取り組みである⁽²⁾。この V-model と MBD 開発を合わせた MBD V-model (Fig. 1) を活用することで、開発の初期工程で懸案事項を排除し、後工程での手戻りがない開発が可能となる。Fig. 1 に車両システム

全体の MBD V-model を示す。Requirements は顧客要望や市場動向、法規などの車両全体に対する要求であり、システム設計者はそれに応じた目標性能を定め、その性能を満たすようにサブシステムや部品の設計をおこなう。近年この取り組みは、各自動車メーカーにも普及し、弊社のパワートレインシステムのデバイス開発においても、この MBD

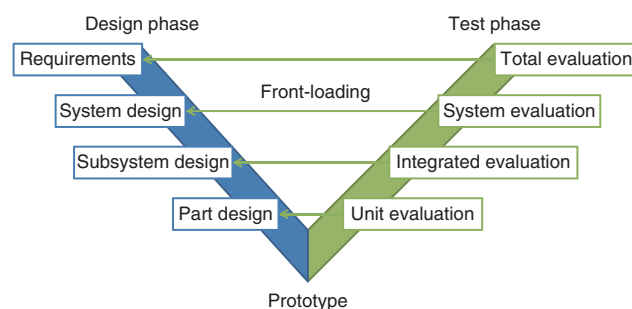


Fig. 1 MBD V-model

*四輪 R&D センター

V-model を用いて開発の効率化が進められてきた^{(3),(4)}。

MBD V-model を車両システム全体に適用していくためには、さまざまな要求から生じた複数の異なる目標性能を整合し、サブシステムや部品の設計をおこなう必要がある。つまり、各サブシステムや部品の設計担当者が各自で効率的な開発をおこなっていても、それらを統合したシステムの目標性能を満足できなければ、そこで手戻りが発生する。

Fig. 2 に従来の車両システム開発イメージを示す。車両全体としての目標性能を検討する会議において、新機種プロジェクトのリーダー（システム設計者）が各サブシステム領域の代表者（PIC）を招集する。そこで各 PIC が、これまで各々が検討してきたサブシステムの目標性能値を提示する。しかし、システム全体の目標性能が未達、又は他のサブシステム性能とトレードオフ関係が発覚した場合は、各 PIC が提示した目標性能値は保留され、各サブシステム領域の担当者達は次の会議までに目標性能の再検討をおこなう。このような手戻りを何度も繰り返すことが多々ある。

このような従来システム開発における課題の原因は、システム全体として目標性能を達成するための技術的な知見と、その関係性が整理できていないことにある。達成見通

しが不明な目標値はただの努力目標であるため、開発期間が続く限り試行錯誤を繰り返すこととなる。逆に、始めから達成の見通しが立てられれば、システム設計者は達成可能なパレート最適解の中から目標性能を戦略的に取捨選択し、サブシステム担当者に対して必要な性能を要求することができる。

このように、手戻りが多い従来システム開発の課題を解決するため、開発の初期段階からシステムの目標性能やそれを達成する手段を定めることができる、以下の開発プロセスを提案する。

- (1) 合理的な開発順序を決める
- (2) システムの目標性能達成可否を予測する
- (3) 初期検討構成で目標性能が未達の際、他の解を探索する
- (4) 複数の解候補から、素早く合理的な解を抽出する

2. 開発プロセスの概要

本研究で提案するプロセスの全体概要を Fig. 3 に示す。まず合理的な開発順序を決めるため、3 章で解説する手法を用いて、対象システムの目標性能とそれを構成する各サブシステムの諸元や特性の関係を整理する。各構成要素とその依存関係をモデル化し可視化することで、構成要素の決定順序が分析でき、プロセスの改善提案が可能となる。

次に、システムの目標性能達成可否を予測するため、整理された関係性の内容を明確化する。4.1. 節で解説する分類に基づき、物理的な関係性のみならず、法規やトレンド、設計手順やタイミング、設計者の知識や経験など、特性を決定するための具体的な方法や指標値を設計者にヒアリングして整理し、シミュレーションができる環境を整える。

しかし、システムやサブシステムの各性能関係が複雑であればあるほど、完全に手戻りのない開発順序を決めることは容易でない。Fig. 3 の手戻り要素群は合理的な開発順序を整理してもなお、手戻りが発生する特性間関係の範

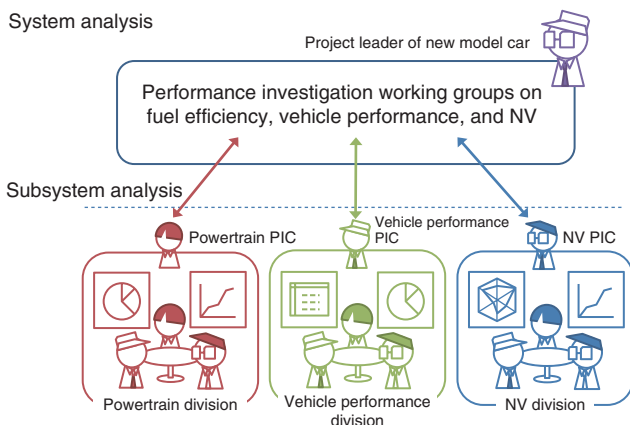


Fig. 2 Complexity of current product development phase

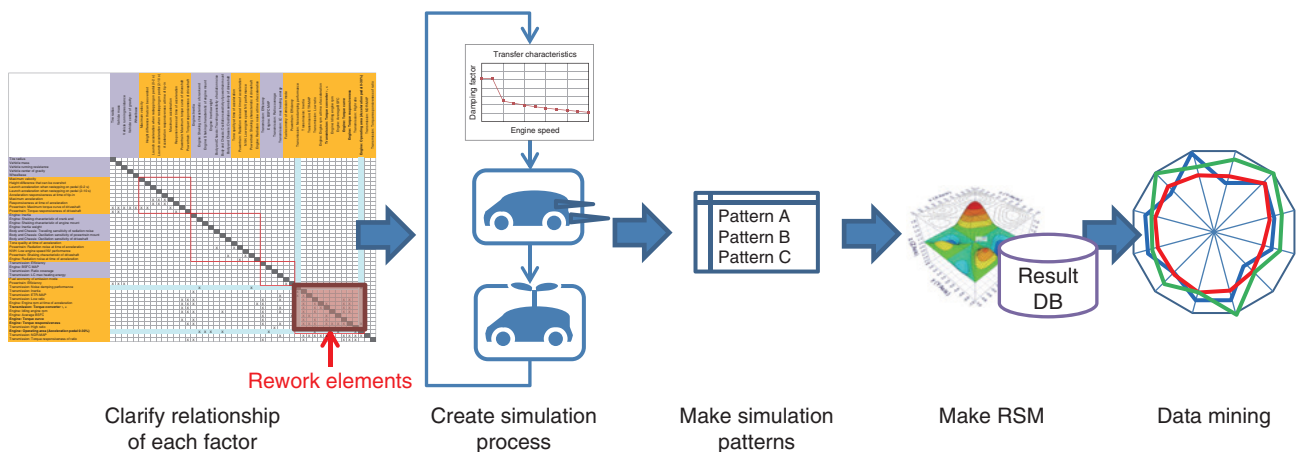


Fig. 3 Process overview

図を示している。手戻りとは、ある性能値や特性値を求める際、計算をおこなう事前条件に自身の特性値を含むという、自己参照関係が存在することを意味する。このような関係でも、Fig. 4 解析的アプローチ上部のように計算ロジックが単純な数式で表現できれば、解析的に解を求めることは可能である。しかし、同図下部のシミュレーションなどの複雑な導出関係が含まれる場合、解析的に解を求めることは容易でない。解が存在しない、または複数存在するケースも発生する。そこで本稿では目標性能が未達の場合でも別の解を探索できるよう、統計的な手法により解探索をおこなう。Fig. 4 の統計的アプローチでは、手戻りが発生する特性間で、手戻り要因となる特性の初期条件（初期値）とその収束条件を設定し、その収束条件を満たす初期値を探索する。ただ手戻り要因が複数存在する場合、その探索範囲は膨大であり、規定の計算時間内で求める解に辿りつける保証がない。また、解がトレードオフになっており自動で解を絞り込めないケースも多々存在する。

そこで、複数の解候補から素早く合理的な解を抽出するため、4.4. 節で解説する有吉ら⁽⁵⁾のデータマイニング技術を用いる。有吉らの手法を用いると、計算結果から高精度な予測を実現する応答曲面モデル (RSM) を作成し、その RSM を用いて短時間で高効率なデータ探索が可能となる。

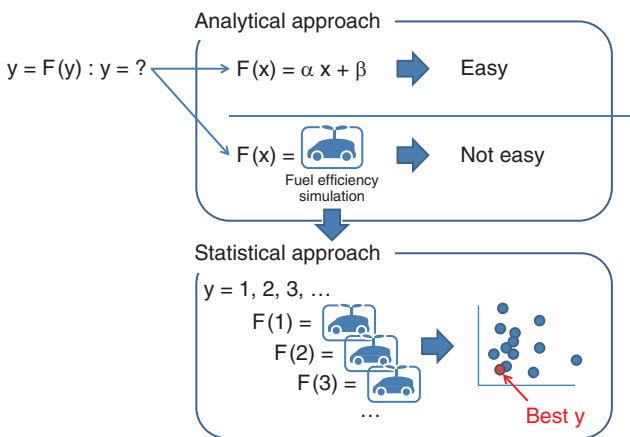


Fig. 4 Approaches toward of solution search

3. DSM を用いた関係性の整理

対象システムを構成する各サブシステムの諸元や特性の関係性を整理するため、Design Structure Matrix (DSM) を使用する⁽⁶⁾。DSM は複雑なシステムの分析と管理の両方を実行するためのシンプルな手法である。DSM を用いて任意システムの各構成要素とその依存関係をモデル化し可視化することで、システム構成の分析ができ、それに伴い各要素の管理も容易になる。また分析結果を基に、各構成要素の決定序列を入れ替えることで、システムの改善提案

が可能となる⁽⁷⁾。

DSM の一般的なモデル化手順を以下に示す。まず対象システムの全構成要素を抽出し、正方行列の縦横にその決定順通りの序列で等しく配置する。その後各列の構成要素が、影響を与えている行要素にその影響依存度を数値で入力する。これを数値 DSM と呼ぶ (Fig. 5(a))。DSM の各構成要素の順列を入れ替えることで、手戻りが少なく、モジュール化された要素順に整理できるため、対象システムの既存の決定プロセスを改善する効果が期待できる。

DSM は、開発に必要なシステムやサブシステムの諸元や特性、目標性能を要素 (エレメント) とするマトリクスである。この各エレメントの設計する順序を整理し、開発する手順を最適化する。本プロセスでは、次章以降で各エレメント間の関係からシステム性能の予測をおこなっていくが、それらの関係は数値 DSM が必要とする影響依存度のような単純な数値で表現することはできない。そのため本稿では単純な依存関係のみをモデル化するバイナリ DSM (Fig. 5(b)) を用い、最適な開発手順を決定することとした。その後、より詳細な関係性は次章で解説するように別途定めた。DSM のモデル化は以下の手順によりおこなった。

- (1) 車両開発に必要なエレメントを、正方行列の縦横に同順に並べる
- (2) 縦軸エレメント (出力) を決めるに当たり、必要な横軸エレメント (入力) に印をつける
- (3) 行列右上半分の印の発生を抑え、かつ車両一台分の性能モジュール化を意識しつつエレメント順序を入れ替える。手戻りが避けられないエレメント群に関しては、できる限りその範囲が小さくなるよう務める
- (4) 並び替えた縦軸エレメントの順番を、そのままエレメントの決定順序と定める

一般的にシステムは、複数のサブシステムから構成され、さらに部品一つにいたるまで階層構造をとる。システムから抽出した各エレメントの階層が大きく異なると、その後の関係性の記述や並び替えが複雑化し、エレメント間の整理が容易ではなくなる。そのため、極力上位および同階層のエレメントを抽出し、その関係性を示す必要がある。

また、DSM で影響度を設定する際、すべての影響を加味して関係を記述することは容易でない。各エレメント間に関係はあるものの、影響度が小さい場合には、DSM の複

Element	A	B	C	D
A				1
B	3		3	
C				
D	2		1	

(a) Numerical DSM

Element	A	B	C	D
A				x
B	x		x	
C				
D	x		x	

(b) Binary DSM

Fig. 5 DSM

雑性を緩和するためあえて関係を外すといった調整が必要となる。さらに影響度が事前に不明なエレメントに関しては、検討初期では関係を考慮するものの、次回以降の検討では関係を外すといった、実業務に合わせたプロセス調整（テラリング）が必要となる。

各エレメントの並べ替えにおいては、絶対的に正しい解は存在しない。The reachability matrix method⁽⁸⁾や Identify loops by path searching⁽⁹⁾などの並べ替えアルゴリズムを用いて、機械的に入れ替えることは可能だが、対象となるシステムにより、モジュール化の考え方や手戻りの許容範囲が異なるため、DSMの並べ替えにはシステム設計者の意思を反映する必要がある。

以上のように、DSMを用いて関係性を整理できたとしても、機械的にエレメントの抽出や関係性付加、最適な決定順番が決まるわけではない。システム設計者や各エレメント担当者が密に連携して、DSMの構築やテラリングをおこなう必要がある。

しかし、DSMを用いて開発すべき各エレメントの設計順序が整理できたとしても、手戻りが避けられないエレメント群が存在する。

4. データマイニングによる解の導出

手戻りが避けられないエレメント群に対して、あらかじめさまざまな案別による予測計算をおこない、複数の解候補の中から、素早く合理的な解に誘導する技術を構築した。具体的には手戻りエレメント群の値を、設計制約の範囲内で広範囲にサンプリングする。その値を初期値としてエレメント群の予測シミュレーションをおこない、得られた複数結果の中から対話型のデータマイニングを用いて解探索をおこなう。

4.1. 各エレメント間の関係定義

システム性能の予測や可視化をおこなうため、各エレメントの関係を明確化しそれに基づきシミュレーションをおこなった。

単純な依存関係のみをモデル化したバイナリ DSM を使って、各エレメント間の影響の有無を示せたが、関係の詳細内容までは表現していない。そこで各エレメント間の関係を明確化するため、以下に示す具体例で分類し、数値的な解を算出するための入出力と計算ロジック、評価基準を定めた。以下に、エレメント間関係の整理に使用した分類の一部を解説する。

制約関係は、法規、彼我比較、トレンドなどの市場要求から目標値として定められるものや、コスト、生産制約などから、設計自由度が小さい、または無いものを指す。これらのエレメントの変更は前提条件の見直しにあたるため、他エレメントに与える影響が大きい。

従属関係は、原理原則が明確になっており、前提条件により一意に定まるものを指す。シミュレーション結果によって定まるものなど、決定アルゴリズムが明確なものも含む。

設計方針および条件付き関係は、シミュレーション結果から逆問題的に定まるものを指す。解が何通りか出現する可能性や、解の選択指針が不明確なものもある。

自由設計関係は、担当者の知識、経験、傾向からおおよそ決まるものを指す。法則化が容易でないものや、抽出項目以外の新たな関係が発生する可能性がある。

以上の例のようにエレメント間の関係を分類することにより、各エレメントの算出根拠が明確になり、計算条件の変更が生じた際も関係者の理解が得られやすい。

4.2. 各エレメント値の計算

DSMによって決定された順番に従い、各エレメント値の算出をおこなう。手戻りが避けられないエレメント群に関しては、その最大ループ構成エレメント内で、実験計画法 (DOE) に基づき複数回計算を繰り返す。手戻りの原因となるエレメントあるいはエレメント群の値を、初期値として設計空間に網羅的にサンプリングし、その値を用いて DSM で決定された順序に沿って、続くエレメント値を先に定めた計算ロジックで計算する (Fig. 6)。

ここで扱う具体的な計算は、各性能領域で事前におこなう検討内容に準拠すべく、簡易な表計算から複数の解析ツールの組み合わせまで、任意のレベルでおこなわれる。

各エレメントは DSM の決定順序に基づき、前段で計算されたエレメント値、および初期値を用いて計算を実施する。

異なる初期値によって計算された各エレメント値を紐付けてデータベース (DB) 化し、目的に応じた分析ツールを用いて可視化をおこなう。

4.3. 最適解の探索手法

DB 化された各エレメントから、システム設計者と PIC 達にとって最適な解の探索をおこなう。最適化の対象となる手戻りエレメント群は、できる限りその範囲が小さくなるよう整理するものの、システム全体の性能検討をおこなう本ケースでは、多入力で多目的の課題になることは避けられない。

多変量多目的な最適解の探索には、解析的なものから統計的なものまで、さまざまなアルゴリズムが提案されている⁽¹⁰⁾。しかし 4 次元以上の多目的な最適化に関しては、解析的に現実的な計算回数で最適解を求めることは容易でない⁽¹¹⁾。

また実際の設計では、数値化することが容易でない暗黙知も踏まえてすり合わせがおこなわれる。そのため、最適解の探索や絞り込みを自動的に進める従来手法では、PIC 達の理解を得ることは容易でない。

そこで本稿では、解析的または統計的に最適解を求める手法ではなく、システム設計者や PIC 達が Graphical User Interface(GUI)を介して対話的に、DB 化された各エレメントの中から求める解を導く、データマイニング技術を使用した。

4.4. 対話型データマイニングによる解探索

Fig. 7 に有吉らが提案する対話型データマイニングの流れを示す。データ探索は以下の手順で進められる。

- (1) 先に作成した計算結果の DB に対して、RSM を用いて設計空間内のデータを近似推定し、解の密度を高める。

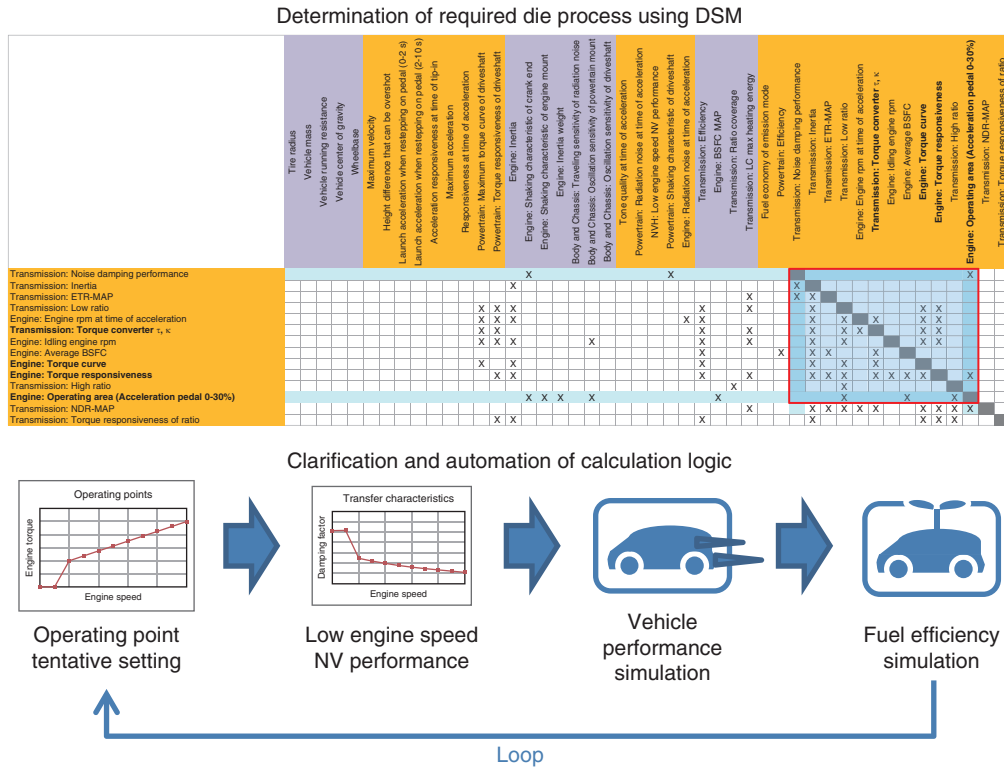


Fig. 6 Coordination process of DSM and simulation

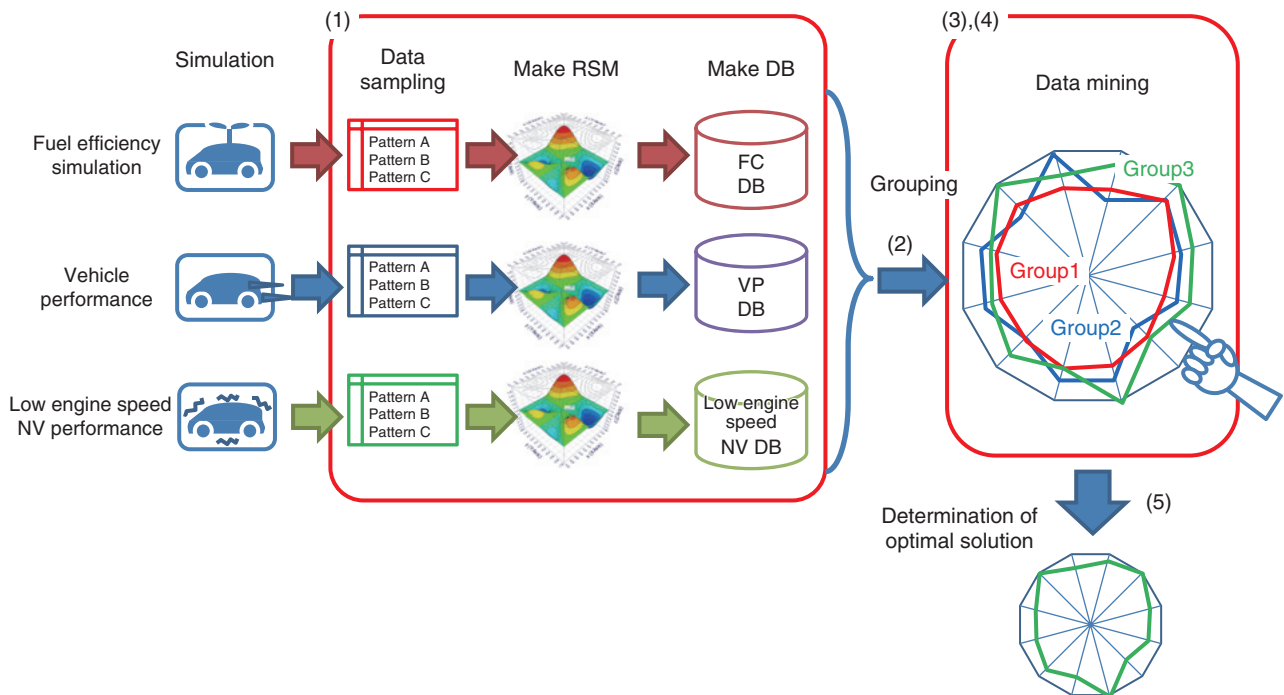


Fig. 7 Interactive solution search

整理されたDSMを参考に、具体的な性能設計をおこなった。本検討では対象領域を限定するため、以降のプロセスにおいてエンジン特性は固定として扱った。手戻り領域のエLEMENTには、動力伝達の中核を担うトランスミッション領域の諸元や制御仕様が多く含まれた。また、手戻りELEMENTはエンジンの推奨運転点(エンジン回転数、エンジントルク)、トルクコンバータ特性(τ , κ)、ファイナルギヤレシオの三つに定まった。

Fig. 10 はトランスミッションの減衰性能を計算するロジックの一例である。エンジンベンチで計測した筒内圧を基に、クランク端のトルク変動を算出する解析モデルを用いて、事前に入出力の数式近似モデルを作成した。この近似モデルをクランク端加振特性と定めた。

Tire radius Vehicle mass Vehicle running resistance Vehicle center of gravity Wheelbase	Maximum velocity Height difference that can be overshoot Height acceleration when resting on pedal (0-2 s) Height acceleration when resting on pedal (2-10 s) Acceleration responsiveness at time of lip-in Maximum acceleration Responsiveness at time of acceleration Reaction time, Maximum torque curve of driveshaft
--	---

評価基準は、算出された減衰性能が本適用機種で使用可

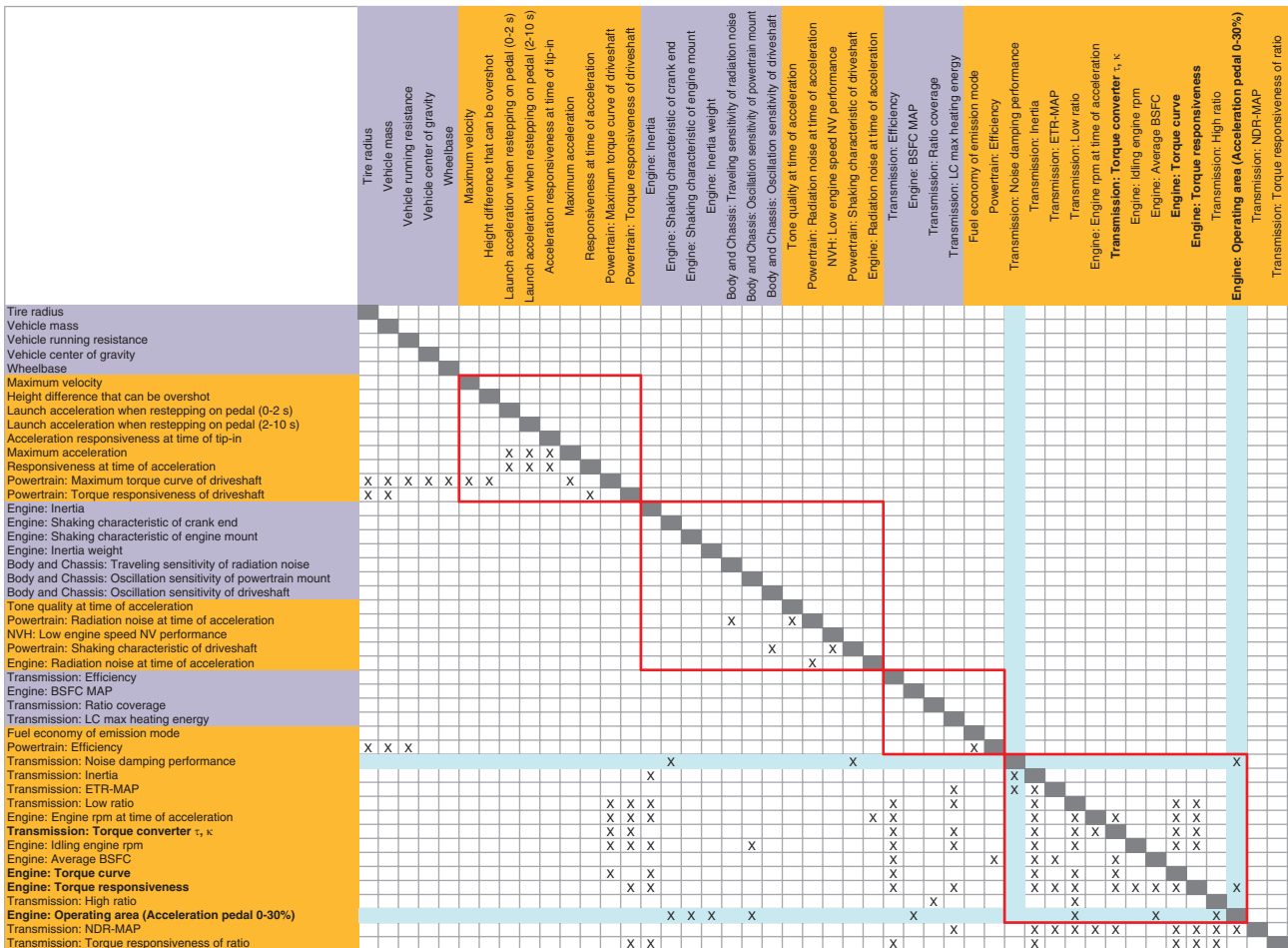
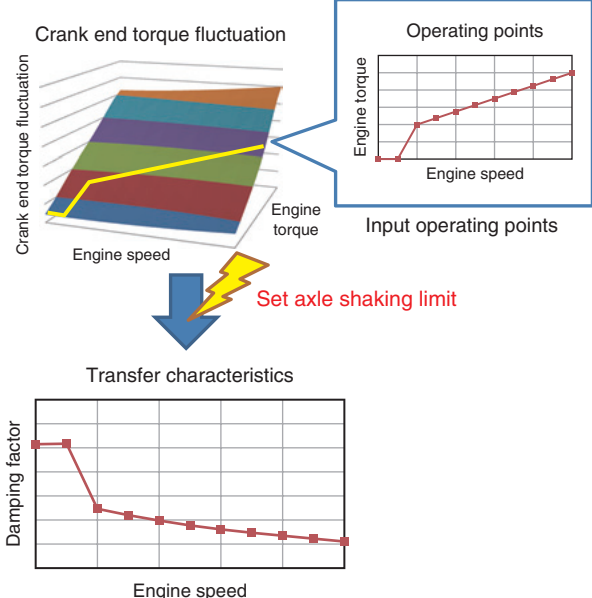


Fig. 9 DSM data after sorting

能なトルクコンバータの減衰率を超えないこととした。この計算ロジックをプログラムで実装し、計算モジュールとして保存した。

手戻り領域の各エレメントに対して上記と同様に計算モジュールの作成をおこない、それらの入出力を繋げて計算をおこなった。手戻りの三つのエレメントは設計空間内で初期値を網羅的にサンプリングした。手戻り領域内をDSMの順番に従って一回ずつ計算した結果を1サンプルとし、約400サンプルの計算をおこなった。計算結果の一部をFig. 11に示す。

Fig. 11に一例として、燃費性能、発進加速、低回転こもり音に関わるエレメントがお互いに影響を及ぼしあい、トレードオフ関係にあることを示す。Fig. 11(a)は縦軸にJC08モードでの燃費性能、横軸に0-100 km/h間の全開発進加速時間を表している。このグラフでは左上に行く程燃費が良く加速時間が短いため、良い性能であることを示している。

但し、その良い結果を示しているサンプルも複数存在するため、その中の2点を抽出し紐づく減衰性能をプロットしたものをFig. 11(b)に示す。この図では縦軸が減衰率、横軸がエンジン回転数を表している。同じ燃費、発進加速でも減衰性能が異なっており、使用できるトルクコンバータが異なることがわかる。

また、この点での加速特性をFig. 11(c)に示す。この図では縦軸にトルクコンバータの特性、横軸に加速応答性(全開発進加速時の0.5 sec後の加速度)を表しており、右にいくほど良い性能であることを示している。抽出した二

つのサンプルは燃費と加速時間においては最も優れているが、加速応答性の観点では両サンプル以上に優れた別のサンプルが存在することがわかる。

このように、レベルが異なるシミュレーション結果をさまざまな観点で確認することで、システム設計者やPIC間で性能に関する議論をおこなうことができる。しかし、各サンプル点に対し、都度個別に結果を比較する方法では、最適な解を見つけるのに時間がかかる。

そこで、最後にデータ選好型のデータマイニング技術を適用した。計算した約400サンプルの計算から高精度なRSMを作成し、それを用いて高効率なデータ探索をおこなった。データ探索により絞り込まれる過程をFig. 12に示す。左から右に進むに従い、解空間の中から最適な解を絞り込んでいる様子を示している。上部のレーダチャートには、燃費や発進加速、使用するトルクコンバータの種類など、一緒に確認したい代表的な性能や特性を示している。

下部の2D図はある性能AとBのみ切り出してプロットしたものである。数あるサンプル点を自動的にクラスタリングし、その代表値がレーダチャートに表示される。左図では、まず初めに全解空間をクラスタリングした結果、Group 1とGroup 2の二つが提示された。そこで、Element EとElement Fの値が小さいGroup 2を選択すると、次は選ばれたGroup 2の解のみで再度クラスタリングをおこない、中央図の二つのグループが提示された。

この作業を繰り返した結果、右図のようにほぼ同等の性能や特性をもつ解のみが残った。このように、システム設計者やPIC達は各性能値のバランスを見ながら、望む傾向

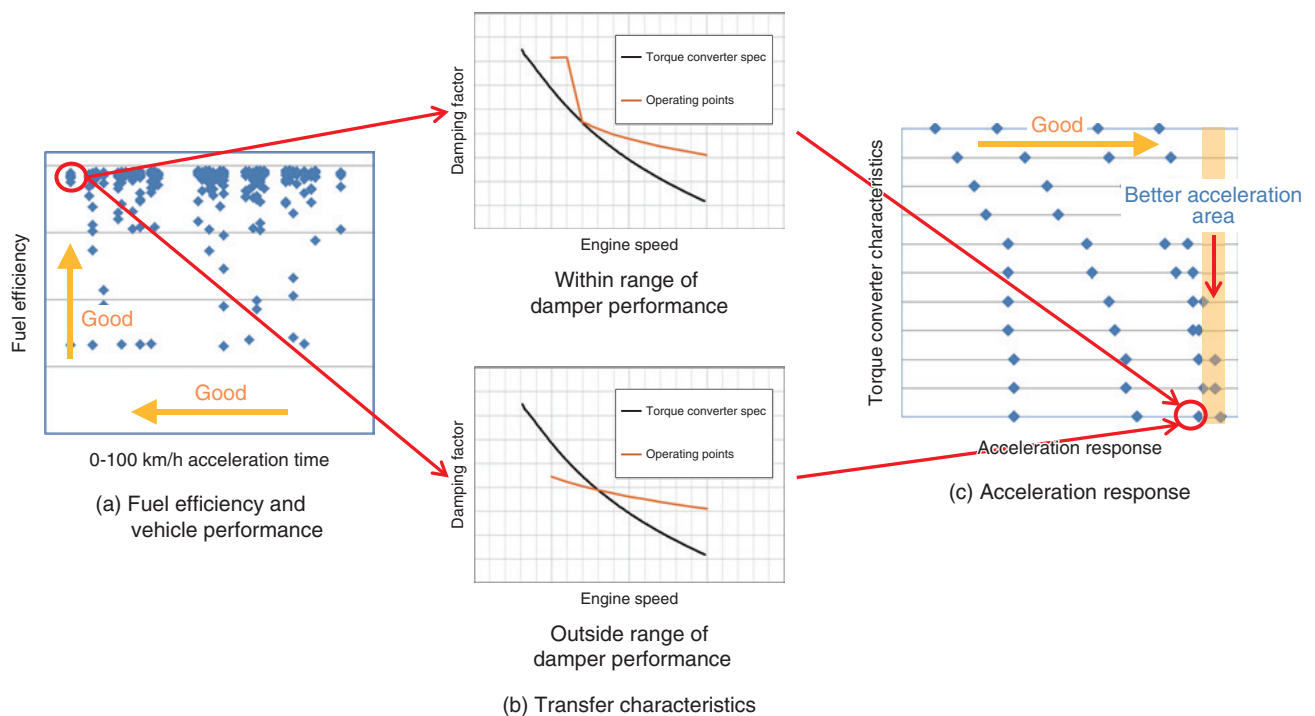


Fig. 11 Example of tradeoff relationship

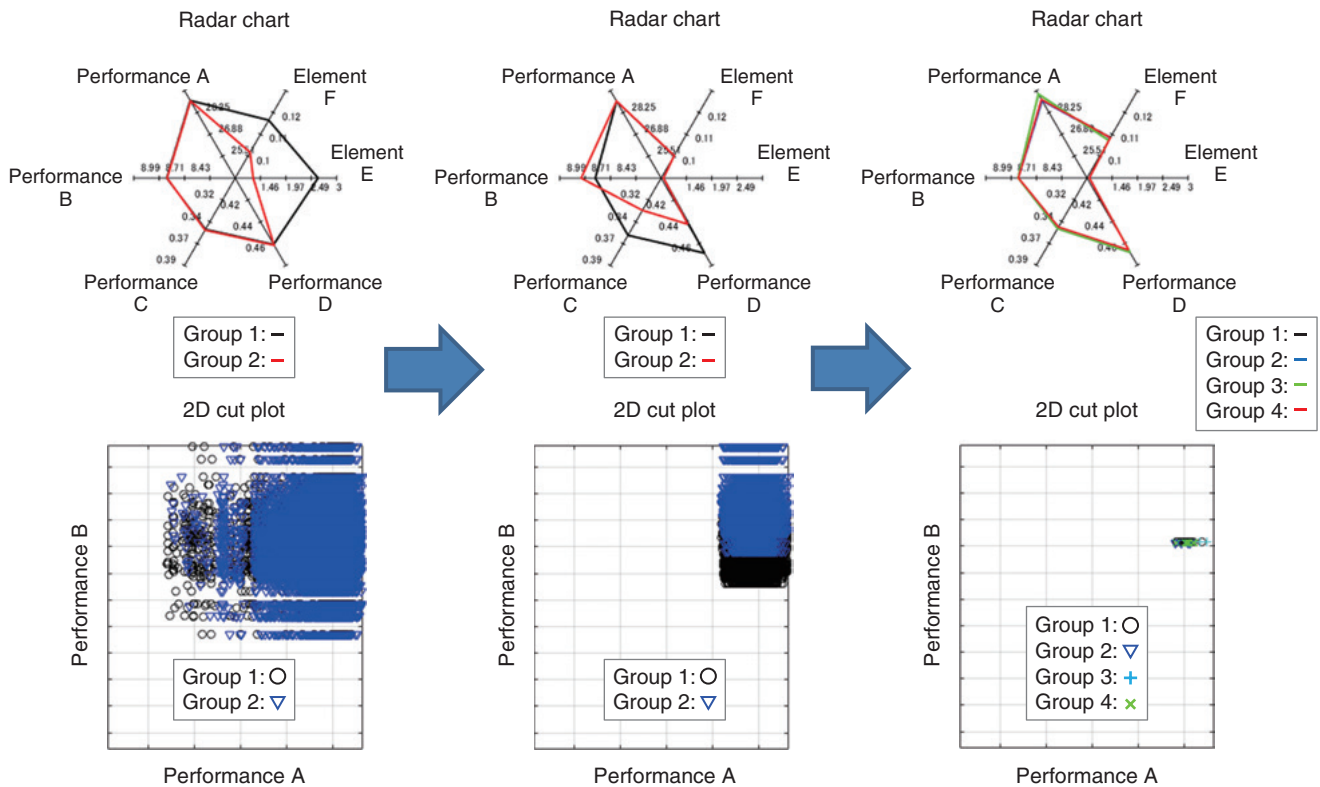


Fig. 12 Data mining process

のレーダチャートを選択し続けることで、解空間の中から最適なバランスの解に絞り込むことができた。

ただ、開発の初期段階ではまだ不確実な要因も多く、特性も変わりやすい。シミュレーションにより事前に整合した初期段階での目標性能が、そのまま実部品で達成できるとは限らない。そのため、初期に目標値を整合する段階でも、ロバスト性を持たせた解の設定が必要である。

以上のように、パワートレインシステム開発の初期段階で燃費、発進加速、低回転こもり音領域の性能影響を整理して計算し、データマイニング技術を用いてシステムの目標性能やそれを達成する手段を定めることで、手戻りの無い開発プロセスを提案できた。

6. まとめ

開発の初期段階からシステムの目標性能やそれを達成する手段を定めることで、手戻りの無い要求型開発をおこなうため DSM やシミュレーション計算、データマイニング技術を組み合わせた、以下の開発プロセスを提案した。

- (1) DSM を用いてシステムやサブシステムの性能や特性間の関係を明確にし、合理的な開発順序を決めた
- (2) 性能や特性間の関係の計算ロジックを定め、システムの目標性能達成可否を予測できるようにした。
- (3) (1), (2) に基づき、手戻り発生エレメントに対して広い設計空間でシミュレーションをおこなうことで、

初期検討構成で目標性能が未達の際でも他の解を探索可能にした

- (4) 膨大な計算結果のマイニング技術を取り入れ、複数の解候補から素早く合理的な解を抽出できるようにした
- パワートレインシステム開発を題材に、燃費性能、発進加速、低回転こもり音領域の性能検討をおこない、システム設計者と PIC 達が最適なバランス解に絞り込めることを検証した。

参考文献

- (1) Forsberg, K., Mooz, H.: The Relationship of Systems Engineering to the Project Cycle, First Annual Symposium of the National Council On Systems Engineering, (1991)
- (2) Schatz, B., Pretschner, A., Huber, F., Philipps, J.: Model-based Development of Embedded Systems, Advances in Object-Oriented Information Systems, p. 331-336, (2002)
- (3) 独立行政法人情報処理推進機構:「組込みシステムの先端モデルベース開発実態調査」調査報告書, p. 5-18, (2012)
- (4) 藤本茂希, 大高義行, 小堂智史: MBD 実現に向けたエンジンプラントモデルと段階的システム同定環境の構築, Honda R&D Technical Review, Vol. 24, No. 2,

p. 120-127

- (5) 有吉斗紀知, 春松正敏, 矢野和重, 福重貴浩, 山田健太郎, 鳴川要: モデルベース開発に向けたデータマイニング技術による車両性能設計システムの研究, Honda R&D Technical Review, Vol. 29, No. 1, p. 178-186
- (6) Steward, D. V.: The design structure system: a method for managing the design of complex systems, IEEE Trans Eng Manage, Vol. 28, No. 3, p. 71-74, (1981)
- (7) Danilovic, M., Browning, T. R.: Managing complex product development projects with design structure matrices and domain mapping matrices, International Journal of Project Management, Vol. 25, Issue 3, p. 300-314, (2007)
- (8) Warfield, J. N.: Binary matrices in system modeling, IEEE Trans Syst Man Cybern, Vol. 3, No. 5, p. 441-449, (1973)
- (9) Gemala, D. A., Eppinger, S. D.: Methods for analyzing design procedures, ASME Design Theory and Methodology, Vol. 31, p. 227-233, (1991)
- (10) Pratab, A., Deb, K., Agrawal, S., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-ii, In KanGAL report 200001, Indian Institute of Technology, Kanpur, India, (2000)
- (11) Laumanns, M., Deb, K., Thiele, L., Zitzler, E.: Scalable test problems for evolutionary multi-objective optimization. tik-technical report, no. 112, p. 16-25, (2001)

■ 著 者 ■



藤本 茂希



東上 圭司



楠 祥治



齊藤 徹