

橋梁パラメトリックモデル  
標準化検討小委員会  
成果報告書

令和3年7月

社会基盤情報標準化委員会

橋梁パラメトリックモデル標準化検討小委員会

# 目次

第 1 章	活動概要 .....	2
1-1	小委員会の名称 .....	2
1-2	検討テーマ .....	2
1-3	検討体制 .....	2
第 2 章	活動の背景と目的 .....	3
第 3 章	パラメトリックモデリングによる橋梁の 3 次元モデル作成指針 (案) .....	4
3-1	本指針の位置付け .....	4
3-1-1	背景と目的 .....	4
3-1-2	3 次元モデルの利用場面 .....	5
3-1-3	対象とする橋梁 .....	6
3-1-4	テンプレートとして定義すべき部位 .....	6
3-1-5	詳細度 .....	7
3-2	パラメトリックモデルの定義 .....	8
3-2-1	パラメトリックモデリング .....	8
3-2-2	テンプレートの定義 .....	8
3-3	CAD ソフトを用いた 3 次元モデルの構築手法 .....	18
3-3-1	CAD ソフトを用いたパラメトリックモデリングによる 3 次元モデルの構築 .....	18
3-3-2	パラメトリックモデルを用いた 3 次元モデルの構築の具体例 .....	19
3-4	組合せ最適化アルゴリズムによる 3 次元モデルの構築手法 .....	32
3-4-1	構築手法 .....	32
3-4-2	組合せ最適化アルゴリズムによる 3 次元モデル構築の具体例 .....	36
3-5	ISO10303 に準拠したデータ構造 .....	45
3-5-1	適用する ISO10303 の仕様 .....	45
3-5-2	操作履歴 .....	56
3-5-3	スケッチ .....	66
3-5-4	アセンブリ .....	75
3-5-5	属性 .....	78

# 第1章 活動概要

## 1-1 小委員会の名称

橋梁パラメトリックモデル標準化検討小委員会

## 1-2 検討テーマ

維持管理を基軸とした橋梁のパラメトリックモデリングの標準化検討

## 1-3 検討体制

委員長／	窪田諭	関西大学環境都市工学部	教授
委員／	塚田義典	摂南大学経営学部	准教授
	梅原喜政	関西大学先端科学技術推進機構	特別任命助教
	中原匡哉	大阪電気通信大学	講師
	今井龍一	法政大学デザイン工学部	教授
	中村健二	大阪経済大学情報社会学部	教授
	山本雄平	関西大学環境都市工学部	助教
	神谷大介	琉球大学工学部	准教授
	姜文淵	関西大学先端科学技術推進機構	特別任命准教授
	井上晴可	大阪経済大学情報社会学部	講師
	関谷浩孝	国土交通省国土技術政策総合研究所	室長
	丸山明	株式会社アイ・エス・エス	代表取締役
	下庄孝弘	アイセイ株式会社	主任
	戸口伸二	株式会社エイテック	リーダー
	三上卓	株式会社エイト日本技術開発	研究サブリーダー
	平見勝洋	株式会社駒井ハルテック	執行役員
	清水智弘	ジェイアール西日本コンサル タンス株式会社	係長
	木村泰三	全日本コンサルタント株式会社	次長
	馬場彰文	株式会社長大	マネージャー
	其田直樹	株式会社 NIPPO	係長
	平野順俊	株式会社日本インシーク	部長
	安井嘉文	株式会社パスコ	課長
	松田克巳	株式会社フォーラムエイト	執行役員

## 第2章 活動の背景と目的

国土交通省は CIM や i-Construction を推進しており、設計・施工段階だけでなく、維持管理に向けて 3次元データの流通基盤の重要性が高まっている。本小委員会は、高架道路橋や鉄道橋をはじめとする橋梁を対象とする。その維持管理においては、次の課題が挙げられる。

課題 1：維持管理が主となる既設構造物への対応が困難である。

- ・設計図や完成図が紛失している構造物やこれらが現況に即していない場合がある。
- ・設計段階からのプロダクトモデルを整備し直すことは非現実的である。
- ・設計段階で作成された 3次元データは、維持管理に全てを引き継ぐとは言い難い。

課題 2：点検データを 3次元データに紐付ける作業に手間がかかる。

課題 3：既設の高架道路橋や鉄道橋の 3次元データと属性情報とを用いた維持管理のための仕様がな

そこで、本小委員会では、既設橋梁を MMS (Mobile Mapping System)、地上型レーザスキャナや UAV (Unmanned Aerial Vehicle) 等で計測した点群データからパラメトリックモデリングによって 3次元モデルを構築し、プロダクトモデルの標準規格である ISO10303 (STEP) に準拠して表現するための指針を検討する。

本検討では、点群データを基にして 3次元モデルを作成する点、維持管理を基軸として 3次元モデルと維持管理情報との関連付けや 3次元モデル表記標準 (案) における 3D Annotated Model との親和性を考慮する点、CAD におけるパラメトリックモデリングとの連携のために ISO10303 を採用する点、および、図面が存在しない既設橋梁の 3次元データを容易に構築できる方法を提案する点に意義がある。

## 第3章 パラメトリックモデリングによる橋梁の3次元モデル作成指針（案）

### 3-1 本指針の位置付け

#### 3-1-1 背景と目的

国土交通省は、3次元データの連携による建設生産プロセスの生産性向上を目指して、i-ConstructionやBIM/CIM（Building Information Modeling/Construction Information Modeling and Management）を推進している。BIM/CIMは、調査・計画、設計、施工、維持管理の各業務段階において、3次元データを一元的に共有、活用し、各業務段階での効率化を図るものである。従来の2次元図面では、それぞれの断面に損傷が表示されるため、全体での損傷状況を把握することが困難であるが、BIM/CIMの導入により、対象構造物を立体的に捉えることができ、損傷の位置や連続性、全体の損傷を把握できる。しかし、維持管理業務に3次元データを導入するためには、膨大な量の既設構造物の3次元データを構築する必要がある。また、紙図面などから仕様に準拠した3次元モデルを生成することは容易ではない。詳細な3次元モデルを作成するための多大な作業コストもBIM/CIMの導入を阻害する課題となっている。

本指針では、社会基盤構造物のうち、橋梁を対象とする。我が国には、橋梁が約73万橋あり、そのうち橋長15m以上の橋梁は約15万橋ある。高度成長期以降に集中的に整備した橋梁が今後急速に高齢化し、10年後には建設後50年経過する橋梁が4割以上になることが見込まれる。そのため、膨大な既設橋梁の維持管理に向けて、3次元データを活用する重要性が高まっている。国土交通省では、橋梁の維持管理の効率化及び高度化を図ることを目的として、3次元モデルの作成仕様や表記標準を策定している。維持管理の主体は既設橋梁であり、紙図面等から仕様に準拠した3次元モデルを生成することは容易ではない。CIMやi-Constructionを有効に実現するために、設計・施工段階だけでなく、膨大な橋梁の維持管理に向けて3次元データの流通基盤の重要性が高まっている。

橋梁の維持管理においては、その形状を正確に把握する必要がある。道路橋示方書では、維持管理に必要な設計図の記録や情報を保存することが規定されているが、

1970年代前後に建設された道路橋の図面は紙媒体であることが多く、工事請負契約の履行期間を経過した後に廃棄される。設計図面が存在しない場合には、現地での再測量が必須である。しかし、橋梁の約7割が市区町村管轄であることを鑑みると、全ての橋梁を再測量することはコスト面でも人的資源の面でも困難である。そこで、Mobile Mapping System (MMS) や地上型レーザスキャナ (Terrestrial Laser Scanner) のレーザ計測機器と、無人航空機 (Unmanned Aerial Vehicle : UAV) による空中写真測量で用いられるデジタルカメラを用いて橋梁の3次元形状を計測することが考えられる。一方、これらの手法で取得した点群データは膨大であり、レーザ光の乱反射や画像対応点の誤検出等に伴う計測誤差を含むことが常である。現場では、橋梁を全方向から隈なく計測できるとは限らない。特に、交通量の多い高架道路橋や植栽の繁茂する山間部の人道橋等では、一方向からの計測に限定され、構造物の角の点群データを取得できない場合がある。従来の3次元モデル作成技術では、このような未計測箇所を含む点群データを想定していない課題がある。

本指針では、あらかじめ定義されたテンプレートに対応する寸法値を入力して3次元モデルを構築するパラメトリックモデリングに着眼し、点群データに最も合致するパラメータを導出するモデリング手法を提案する。これにより、対象を隈なく計測できていない点群データであっても、地物の概形を保持した3次元モデルをより精確に生成することを目指す。

本指針では、以下の内容について述べる。

- 橋梁部位のテンプレート
- パラメトリックモデリングによる3次元モデルの作成手法
- ISO10303に準拠したデータ構造
- 3次元モデルの品質（精度、完全性等）と詳細度

### 3-1-2 3次元モデルの利用場面

本指針、つまり、パラメトリックモデリングによる3次元モデルの利用場面として、橋梁点検業務、橋梁補修・補強設計業務におけるベースモデルの作成、及び、現況地形も含めた点群モデルとの併用による施工計画の検討資料を設定する。

### (1) 橋梁点検業務

橋梁点検において、損傷箇所や写真を記録し、点検調書を作成する。本指針の3次元モデルは、図面がない橋梁を対象とし、これら点検を記録する基盤とする。次の利用を想定する。

- 損傷箇所や現場写真を記録するベースマップとして利用する
- 点検調書に掲載する一般図の代替として利用する
- 橋梁維持管理における基礎調書として利用する
- 建設時の図面がない橋梁が多いと考えられ、そのような橋梁においても3次元化することにより、維持管理時の点検計画・点検記録を3次元モデルに保存し活用する
- 橋梁点検車などの運用可否の判断に利用する

### (2) 橋梁補修・補強設計、補修施工におけるベースモデルの作成

前項と同様に、3次元モデルは、補修・補強設計業務や補修施工業務で用いられる基盤データとなり得る。

- 点検・補修設計時の下絵・工事等の履歴確認資料
- 現況地形も含めた点群データとの併用による施工計画の検討資料

## 3-1-3 対象とする橋梁

本指針の対象とする橋梁は、構造が複雑でない橋梁であり、鈹桁橋、T桁橋、床版橋とする。なお、桁が断面変化する場合には、対象とする。

## 3-1-4 テンプレートとして定義すべき部位

パラメトリックモデルで用いるテンプレートとして定義する部位は、地上型レーザスキャナ、MMSやUAV搭載型カメラ等の計測機器により計測が可能であり、計測による点群データから3次元モデルを構築できる部位とする。計測機器によって点群データを取得できない部位は、対象外とする。

### 3-1-5 詳細度

土木分野におけるモデル詳細度標準(案)において、構造物(橋梁)の詳細度(案)が示されており、LOD (Level Of Detail) 100 から 500 の 5 段階で定義づけられている。LOD ごとの利用用途は、調査、設計、製作、施工、維持管理の各事業プロセスに応じて一律に定めるのではなく、モデル化対象毎に必要な詳細度をレベルで区分することとなっている。

本指針では、3次元モデルを橋梁点検において損傷箇所、点検情報や現場写真を記録するベースマップと一般図の代替として利用すること、および、技術者の意見を踏まえて、LOD200 を目標とする。LOD200 は対象の構造形式がわかる程度のモデルである。さらに、LOD300 についても検討する。LOD300 は対象の外形形状を正確に表現したモデルである。



## 3-2 パラメトリックモデルの定義

### 3-2-1 パラメトリックモデリング

パラメトリックモデルは、あらかじめ定義されたテンプレートに対応する寸法値や角度などのパラメータを入力するだけで、容易に作成および修正が可能な 3 次元モデルである。

パラメトリックモデルのイメージを図-1 に示す。図では、パラメトリックモデルのために提案する T 型橋脚の 2 次元スケッチを表し、p1 から p5 がパラメータを示す。従来、3 次元モデルの構築では、汎用 CAD ソフトウェアを用いて端点の座標、距離や角度などを詳細に指定し断面を作成していたが、パラメトリックモデルを導入することにより、土木構造物を選択し、パラメータを入力するだけで 3 次元モデルの作成および修正を行えるようになり、作業が簡略化されるため、作業時間の短縮の効果が期待される。

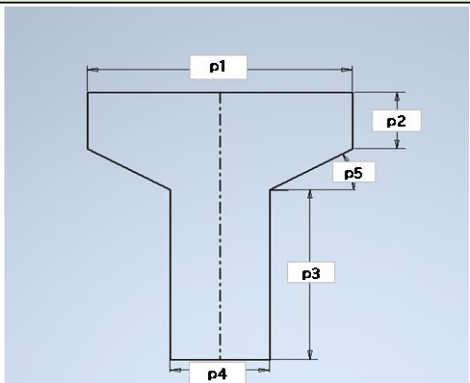
LOD	200	
スケッチ	場所	パラメータ
	p1	寸法値(m)
	p2	寸法値(m)
	p3	寸法値(m)
	p4	寸法値(m)
	p5	角度(°)

図-1 パラメトリックモデル

### 3-2-2 テンプレートの定義

本研究で定義するテンプレートは、平面形状である 2 次元スケッチとする。図-2 に示すように、スケッチに対して、CAD ソフトによる押し出し操作により 3 次元モデルを作成する。パラメータには、設計条件にて設定される寸法、勾配や構造計

算・安定計算などにより算出された寸法を使用することを基本とする。

橋梁のテンプレート作成対象の部位は、主桁、横桁、縦桁、床版、橋脚、橋台である。高欄と支承については、テンプレートとして全ての形状を定義するのではなく、必要に応じて追加する。

提案するテンプレートは、**図-3**に示すように、Autodesk Inventor で作成した 2次元スケッチに Excel の数値を連動させ、Excel の数値を変更すればテンプレートの寸法を編集できるようにする。テンプレートのパラメータ設定は、日本橋梁建設協会のデザインデータブックを参照し、パラメータ設定の自動化のために必要最小限となるように設定した。

本指針におけるパラメトリックモデルのためのテンプレートを**図-4**から**図-15**に示す。テンプレートの対象は、前述の床版、主桁、縦桁、横桁、橋脚、橋台である。T型橋脚と T 型 2 柱式橋脚では、3次元モデルの構築において、寸法の設定が複数ありうるため、角度寸法がある場合と縦と横の長さにより角度を表現する場合の 2種類のテンプレートを作成した。

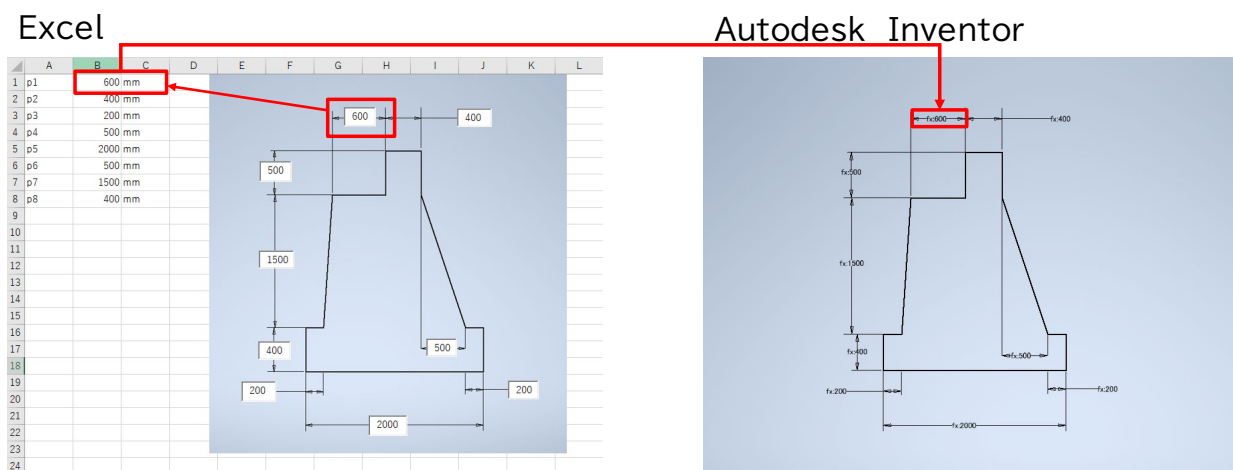
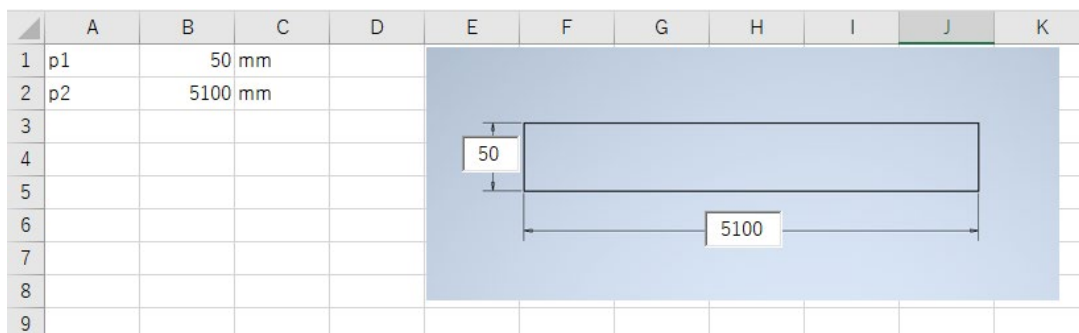


図-3 提案するテンプレート

## 2次元スケッチ



## 3次元モデル

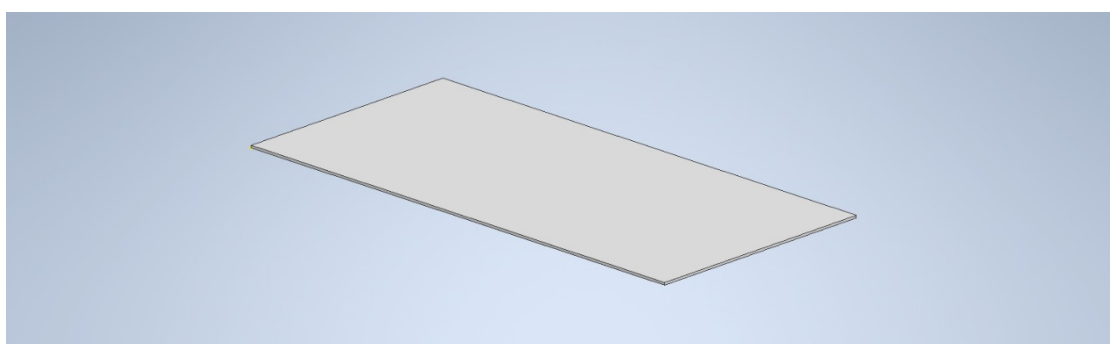
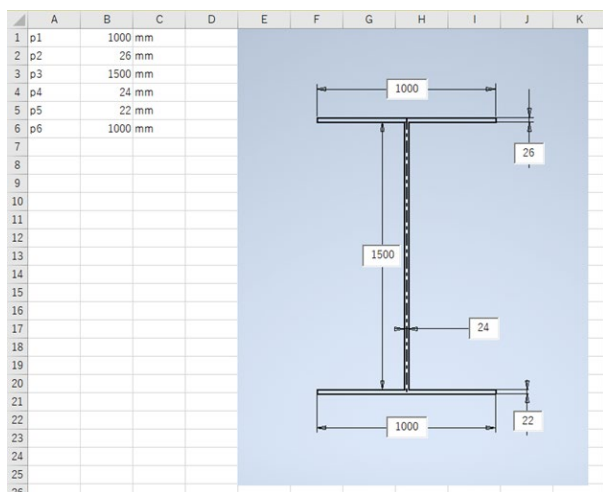


図-4 床版テンプレート

## 2次元スケッチ



## 3次元モデル

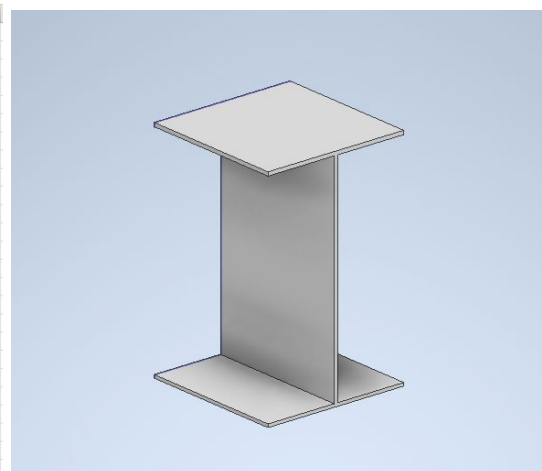
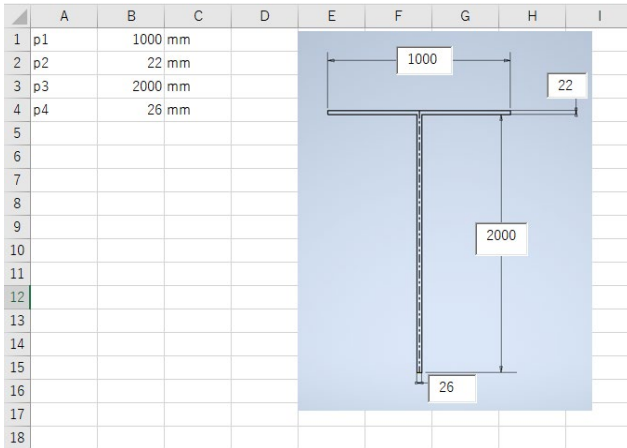


図-5 縦桁、横桁テンプレート

### 2次元スケッチ



### 3次元モデル

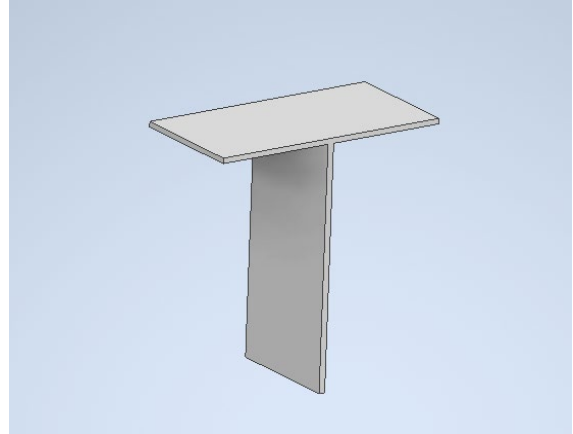
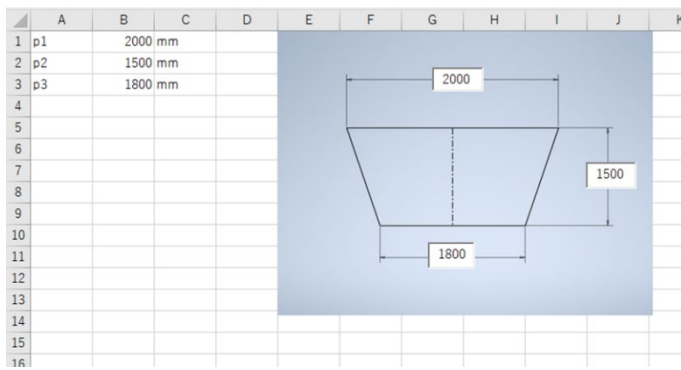


図-6 T桁テンプレート

### 2次元スケッチ



### 3次元モデル

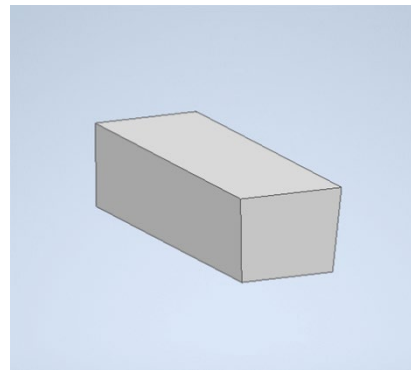


図-7 箱桁テンプレート

## 2次元スケッチ

## 3次元モデル

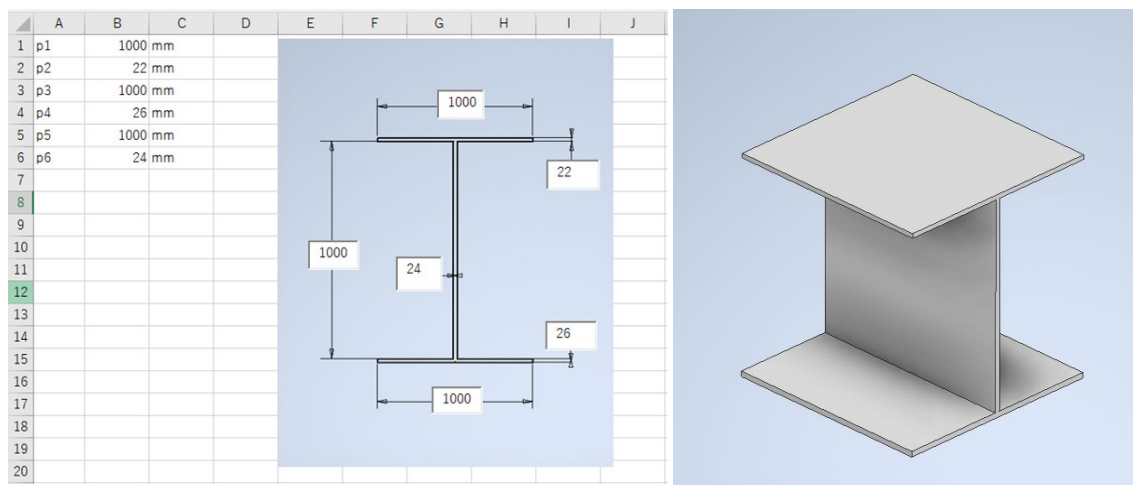
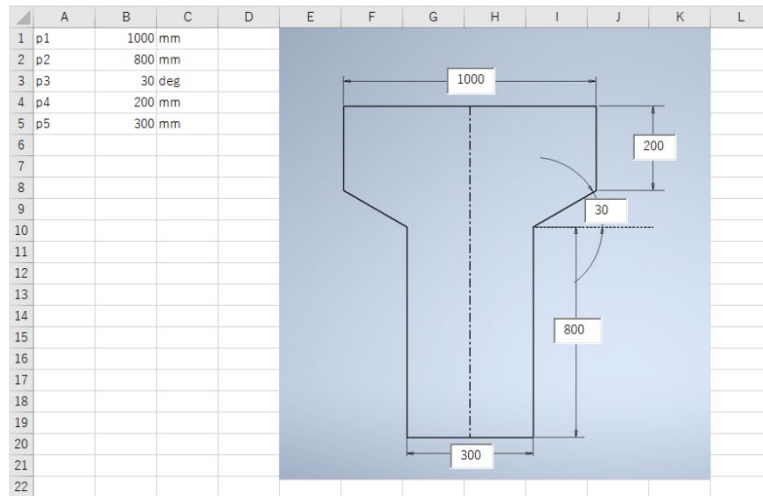


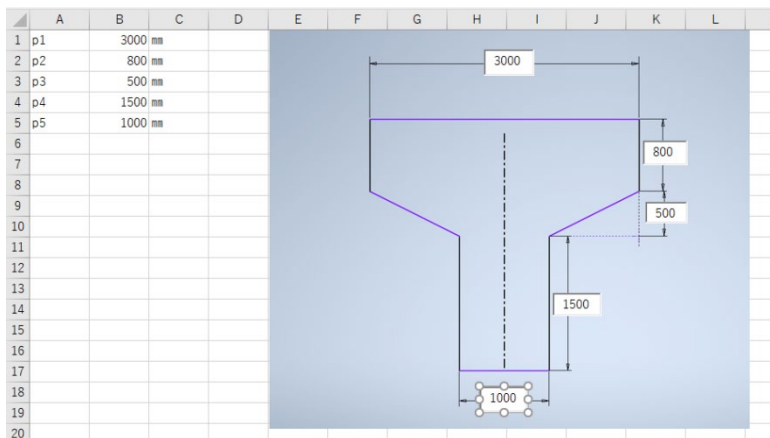
図-8 I桁テンプレート

## 2次元スケッチ

角度寸法あり



角度寸法なし



## 3次元モデル

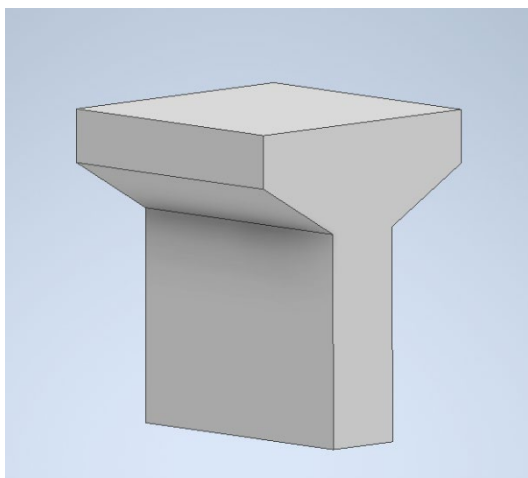
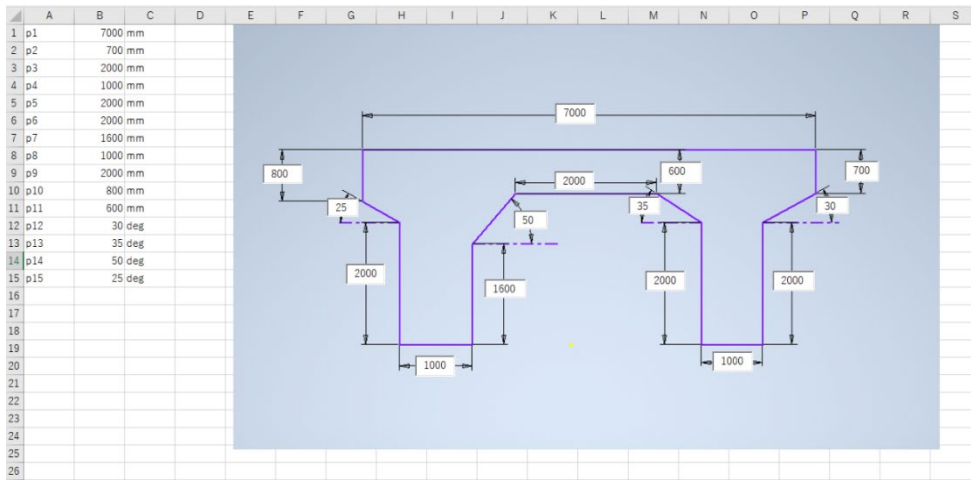


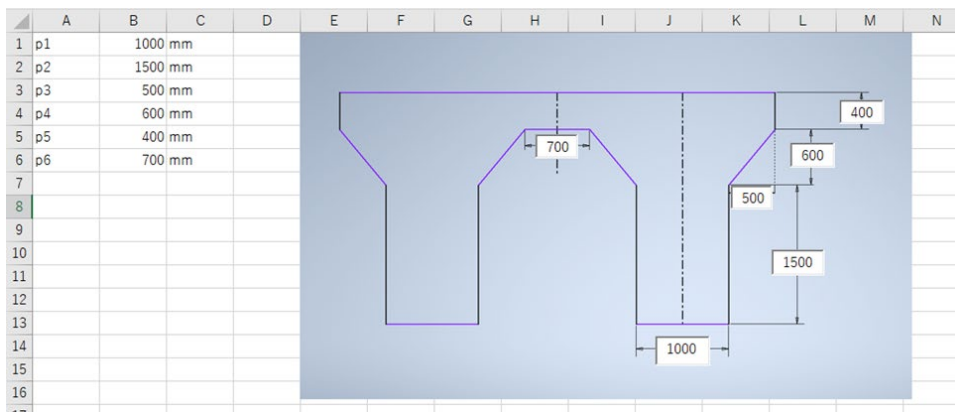
図-9 T型橋脚テンプレート

## 2次元スケッチ

### 角度寸法あり



### 角度寸法なし



## 3次元モデル

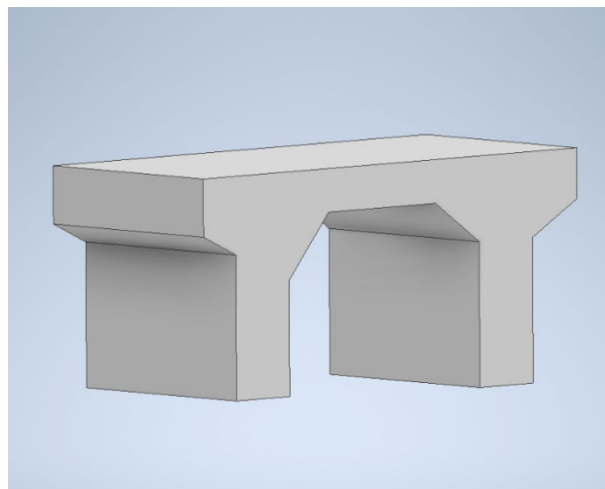
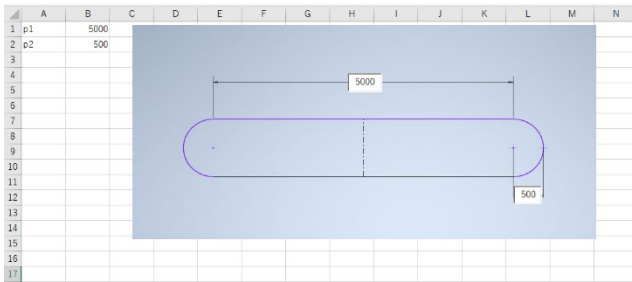


図-10 T型2柱式橋脚テンプレート

### 2次元スケッチ



### 3次元モデル

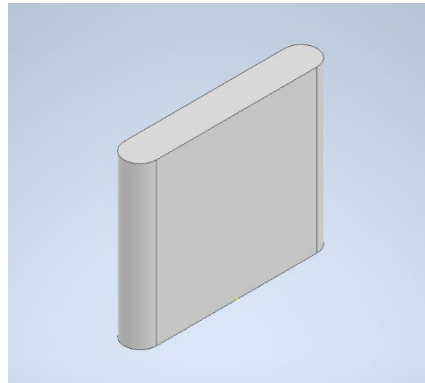
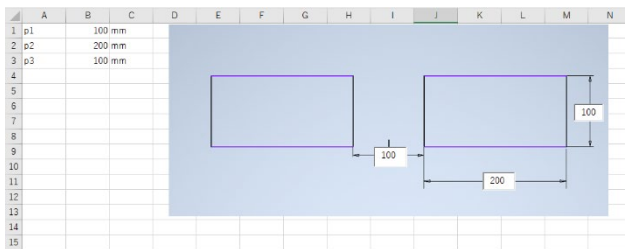


図-11 壁型 1 柱式（小判形）テンプレート

### 2次元スケッチ



### 3次元モデル

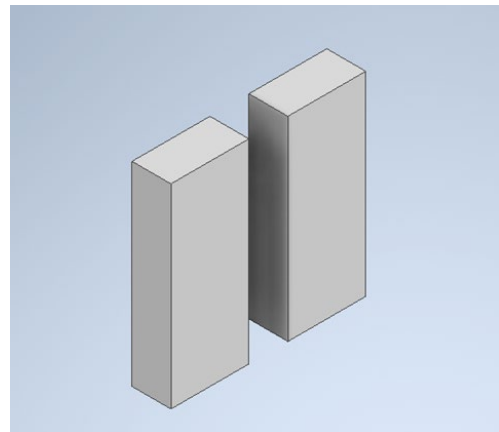
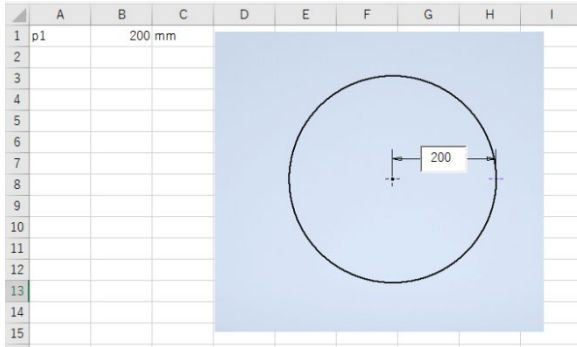


図-12 壁型 2 柱式（矩形）テンプレート



2次元スケッチ



3次元モデル

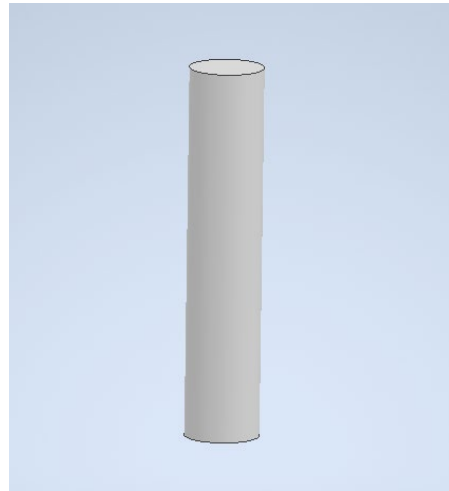
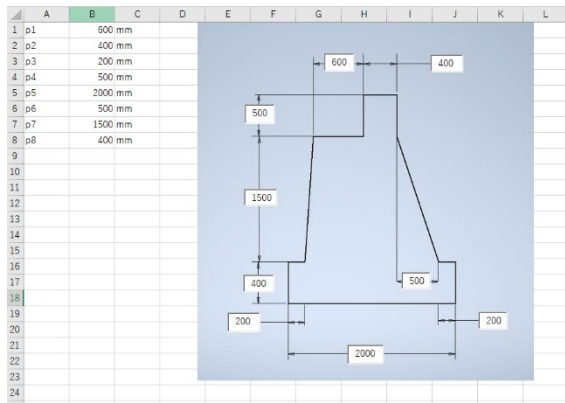


図-13 壁型1柱式(円形)テンプレート

2次元スケッチ



3次元モデル

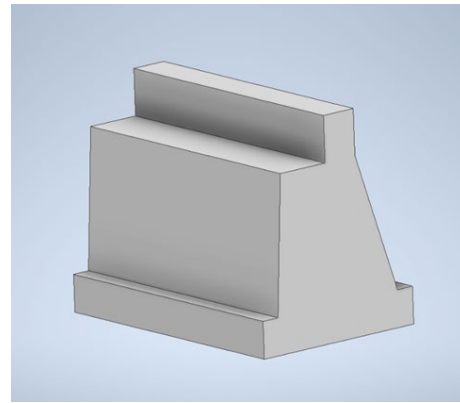
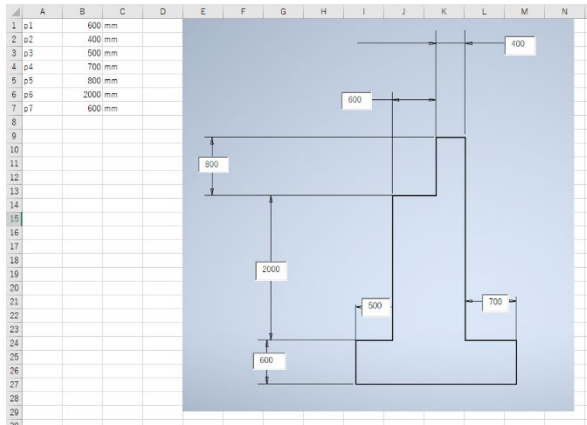


図-14 重力式橋台テンプレート

## 2次元スケッチ



## 3次元モデル

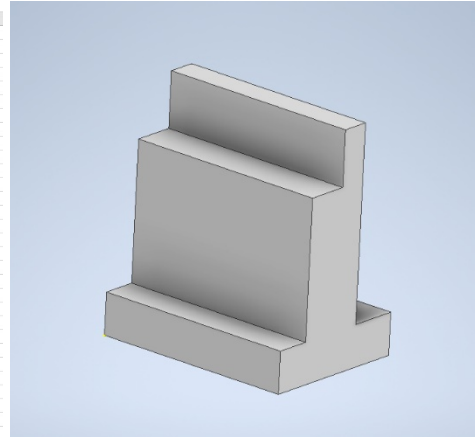


図-15 逆 T 型橋台テンプレート

### 3-3 CAD ソフトを用いた 3 次元モデルの構築手法

#### 3-3-1 CAD ソフトを用いたパラメトリックモデリングによる 3 次元モデルの構築

本指針における CAD ソフトを用いた 3 次元モデルの構築手法を図-16 に示す。

図中①では、対象橋梁を地上型レーザスキャナ、MMS や UAV 搭載型カメラにより計測する。

②では、計測データから点群データを生成する。UAV 搭載型カメラによる空中写真測量を用いた場合は、点群データ生成ソフトを利用して SfM (Structure from Motion) /MVS (Multi View Stereo) 技術により点群データを生成する。生成した点群データは、CAD ソフトで編集可能なファイル形式に変更する。

③では、②の点群データを基に、3 次元モデルの作成に必要なテンプレートとして、部位毎のテンプレート一覧から選択する。3 次元モデル構築者は、選択したテンプレートにおいて、点群データから点間距離を計測して必要なパラメータを取得し、テンプレートの寸法をあらかじめ合わせておく。

④では、③で選択し、寸法を合わせたテンプレートを押し出し操作したパラメトリックモデルと点群データとを手動で位置合わせし、⑤の 3 次元モデルを構築する。

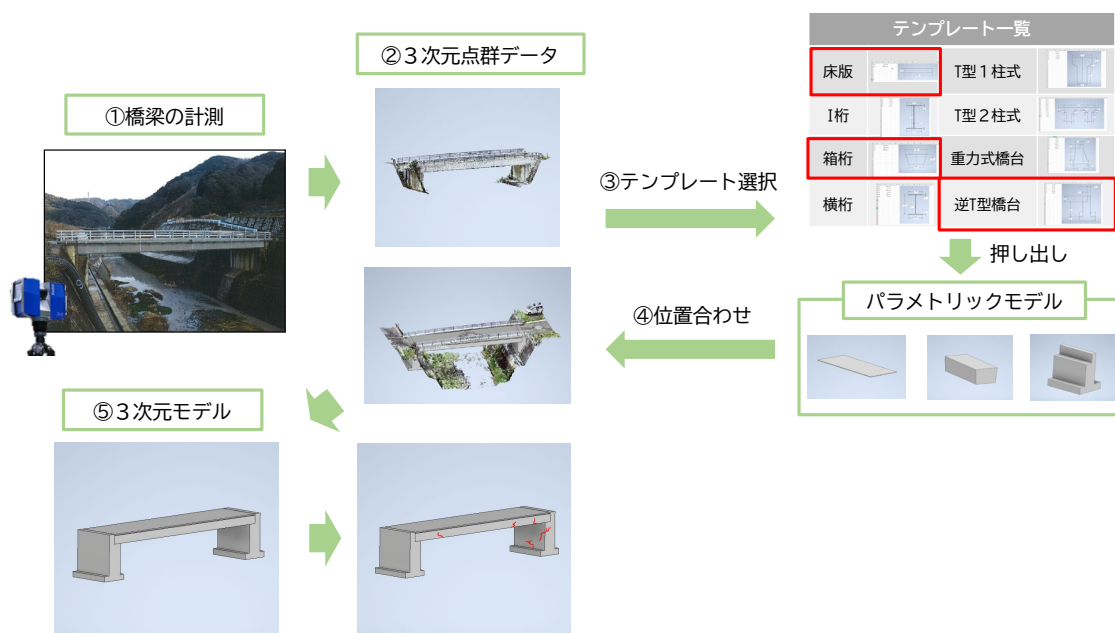


図-16 CAD ソフトを用いた 3 次元モデルの構築手法

### 3-3-2 パラメトリックモデルを用いた 3 次元モデルの構築例

本項では、地上型レーザスキャナ、UAV 搭載型カメラと MMS による点群データを用いて、CAD ソフトを用いてパラメトリックモデリングにより 3 次元モデルを構築する。

#### (1) 構築例 1 (地上型レーザスキャナによる点群データ)

##### 1) 対象橋梁

地上型レーザスキャナ FARO Focus3D X330 による計測データを用いて、橋梁の 3 次元モデルを構築する例を示す。構築には、大阪府泉南市の童子畑橋と大阪府泉佐野市女形橋の点群データを使用した。童子畑橋 (図-17) における計測箇所は図-18 に示す 9 箇所であり、橋梁全体の詳細なデータを得るために、橋梁の上部から 3 箇所 (黄色)、橋梁の下部から 6 箇所 (赤色) で計測した。計測データを用いて作成した点群データを図-19 に示す。女形橋 (図-20) における計測箇所は、図-21 に示す橋梁を中心とする 6 箇所である。計測データを用いて作成した点群データを図-22 に示す。



図-17 童子畑橋



図-19 童子畑橋の点群データ

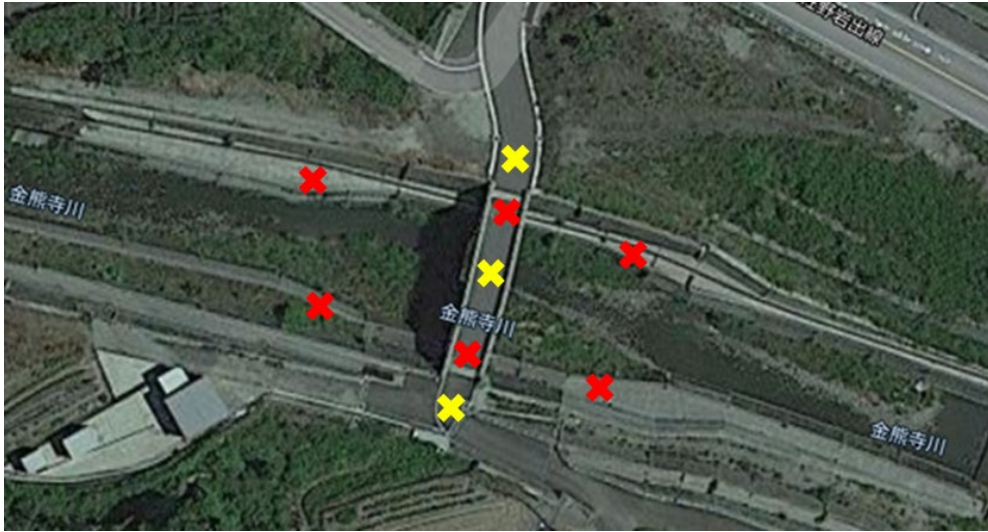


図-18 童子畑橋における計測位置



図-20 女形橋



図-21 女形橋における計測位置



図-22 女形橋の点群データ

## 2) 3次元モデルの構築

図-23 に示す童子畑橋の3次元モデル構築においては、図中(a)の点群データを基に、床版、箱桁、逆T型橋台の3種類のテンプレートを選択した。選択したテンプレートでは、Autodesk Inventorの計測機能でパラメータを取得し、寸法を合わせる。(b)寸法を合わせたテンプレートを押し出し操作したパラメトリックモデルと点群データとを位置合わせし、(c)3次元モデルを構築した。

図-24 に示す女形橋においても、童子畑橋と同様の手順で3次元モデルを構築した。構築においては、(a)点群データを基に、床版、箱桁、T型橋脚、逆T型橋台の4種類のテンプレートを選択した。(b)選択したテンプレートのパラメトリックモデルを位置合わせし、(c)3次元モデルを構築した。テンプレートに定義していない部位に損傷があった場合を想定し、後から3次元モデルにAutodesk Inventorで作成した高欄を追加した。

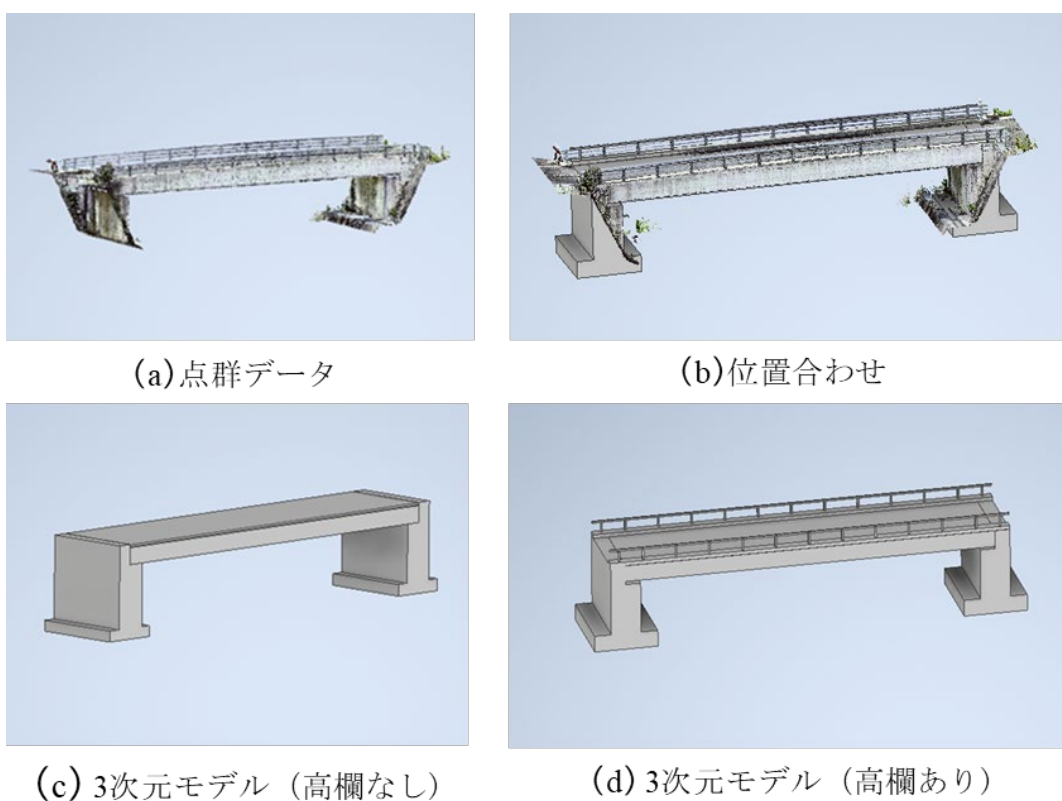


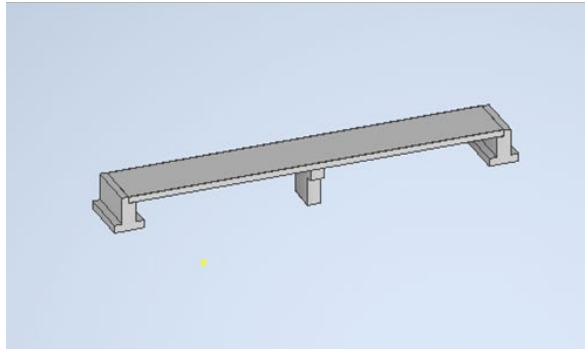
図-23 童子畑橋の3次元モデル



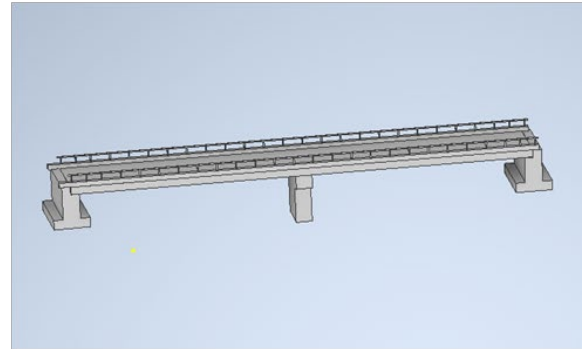
(a)点群データ



(b)位置合わせ



(c) 3次元モデル (高欄なし)



(d) 3次元モデル (高欄あり)

図-24 女形橋の 3次元モデル



## (2) 構築例 2 (地上型レーザスキャナによる点群データ)

### 1) 対象橋梁

地上型レーザスキャナ FARO Focus3D X330 による計測データを用いて、**図-25**に示す桁の構造が複雑である場合に、3次元モデルを構築する例を示す。構築には、大阪府泉南市信達童子畑付近の新前川橋の点群データを使用した。新前川橋 (**図-26**) における計測箇所は**図-27**に示す 8 箇所であり、橋梁全体の詳細なデータを得るために、橋梁上部から 2 箇所 (黄色)、橋梁下部から 6 箇所 (赤色) で計測した。計測データを用いて作成した点群データを**図-28**に示す。



**図-25** 新前川橋の点群データ (下面)



**図-26** 新前川橋

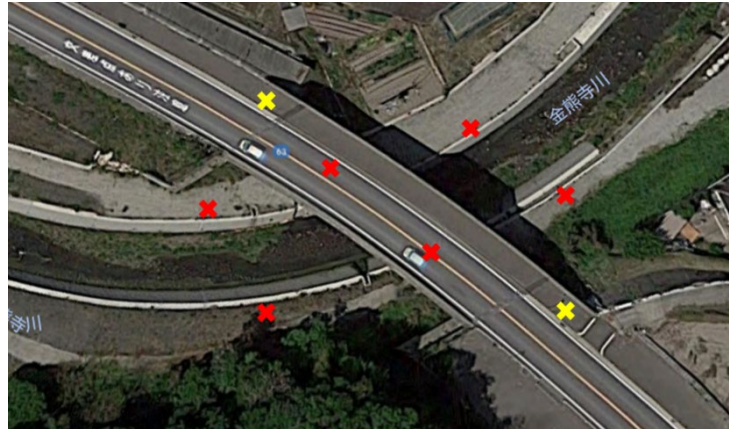


図-27 新前川橋における計測位置



図-28 新前川橋の点群データ

## 2) 3次元モデルの構築

図-29 に示すように、童子畑橋と同様の手順で3次元モデルを構築した。3次元モデル構築には、(a) 点群データを基に、床版、横桁、I桁と逆T型橋台の4種類のテンプレートを選択した。また、テンプレートに定義していない補剛材を追加した。

構築の結果、地上型レーザスキャナは一地点から高精度に計測できるため、橋梁の下から計測可能であれば、桁構造が複雑でも (b) パラメトリックモデルを位置合わせし、(c) 3次元モデルを構築することができた。また、(d) テンプレートにない補剛材を追加し、定義していない部位を含めて3次元モデルを構築できた。

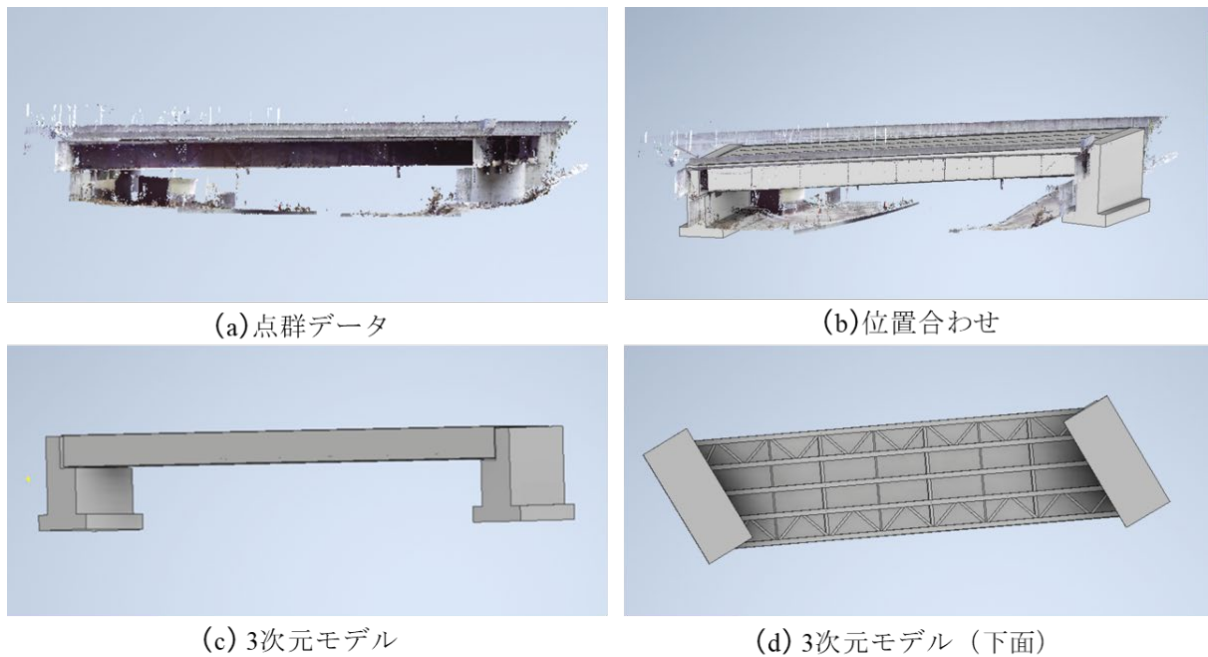


図-29 新前川橋の3次元モデル

### (3) 構築例 3 (UAV 搭載型カメラによる計測データ)

#### 1) 対象橋梁

UAV 搭載型カメラによる画像データを用いて SfM/MVS (Structure from Motion/Multi View Stereo) 処理した点群データから、橋梁の 3 次元モデルを構築する例を示す。構築には、岐阜県揖斐郡揖斐川町の旧落合橋 (図-30) を DJI 社製の Mavic Mini によって撮影した動画から画像を切り出し、Agisoft 社製 Metashape によって SfM/MVS 処理した点群データを使用した。図-31 に UAV の計測経路を示す。カメラは、経路①では下向き、経路②と③では橋梁側に向け計測された。約 2 分の動画をオーバーラップ 80%以上となるよう、1 秒間隔に分割した 146 枚の画像を用いて SfM/MVS 処理した。生成した点群データを図-32 に示す。146 枚の画像のうち 119 枚しか写真のアラインメントができなかったため、点群データに欠損が見られる。



図-30 旧落合橋



図-31 旧落合橋における UAV の飛行経路



図-32 旧落合橋の点群データ

## 2) 3次元モデルの構築

旧落合橋の3次元モデルの構築には、床版、箱桁と壁型1柱式（小判形）の3種類のテンプレートを使用した。構築の結果、橋脚は十分に計測できているため、テンプレートに必要なパラメータをAutodesk Inventorの計測機能で取得し、寸法を合わせた。上部構造では、点群データの図-33(a)左側上部に欠損が見られるが、(b)右側上部の計測できた一部のデータからテンプレートに必要なパラメータを取得し、寸法を合わせた。その結果、UAV搭載型カメラによる計測データを用いて(c)パラメトリックモデルを位置合わせし、(d)3次元モデルを構築した。



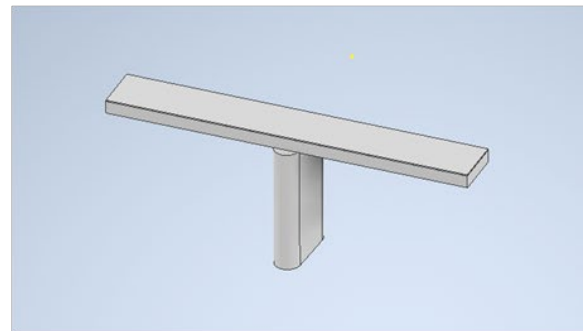
(a)点群データ（左側）



(b)点群データ（右側）



(c)位置合わせ



(d)3次元モデル

図-33 旧落合橋の3次元モデル

#### (4) 構築例 4 (MMS による点群データ)

##### 1) 対象橋梁

MMS による点群データとして、阪神高速道路 11 号池田線高架下に隣接する道路を計測したデータを借用した。高架橋下の道路を走行しながら MMS で計測されたため、一方向からの計測に限定された。計測データを用いて作成した点群データを図-34 に示す。未計測箇所が多く、MMS の取得点数が 10 万点/s と少ない点群データである。



図-34 高架道路橋の点群データ

## 2) 3次元モデルの構築

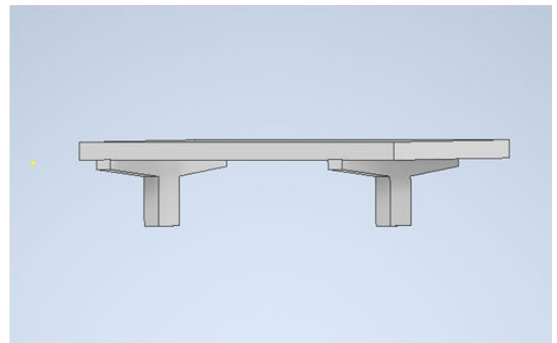
3次元モデルの構築には、床版とT型橋脚の2種類のテンプレートを使用した。構築した3次元モデルを図-35に示す。(a)使用した点群データは、高架橋下の道路を走行しながらMMSで計測され、一方向からの計測に限定された。未計測箇所が多く、点密度が低いので、拡大すると点が粗になり部位を判別できず、全体の点群から部位を判別した。また、取得点数が少ないためテンプレートに必要なパラメータを正確に取得することが困難であった。そのため、(b)選択したパラメトリックモデルを位置合わせし、(c)3次元モデルを構築した。ただし、地上型レーザスキャナによる点群データを用いた場合と比べると精確さに劣る。



(a)点群データ



(b)位置合わせ



(c)3次元モデル

図-35 高架道路橋の3次元モデル



### 3-4 組合せ最適化アルゴリズムによる 3次元モデルの構築手法

#### 3-4-1 構築手法

##### (1) 処理の流れ

本手法は、パラメトリックモデリングにおいて、点群データに最も合致する寸法等のパラメータを組合せ最適化アルゴリズムによって導出するモデリング手法である。パラメトリックモデリングを採用することにより、寸法等のパラメータを適宜変更して、既設橋梁の3次元モデルをより精確に生成する。例えば、橋梁下部工のT型橋脚のパラメトリックモデル(図-36と表-1)は、6つのパラメータと3つの制約条件を保持する。各種パラメータの値は、構造物毎に異なるため定めない。ただし、概形を維持するために、パラメータ間の平行や垂直等の幾何的特徴とパラメータ間の大小関係等の制約条件を持つ。このテンプレートでは、組合せ最適化アルゴリズムを解くために、3-2-2項のテンプレート案のT型橋脚に押し出し量をパラメータとして加えた。つまり、スケッチはp1からp4とp6で構成される平面であり、p5の寸法値だけ押し出して3次元モデルが作成される。

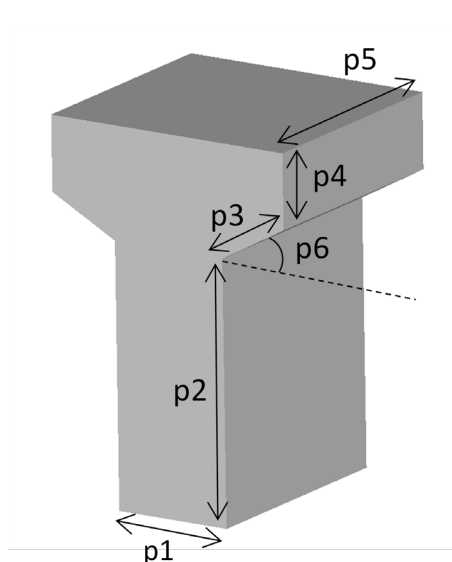


図-36 T型橋脚のテンプレート

表-1 T型橋脚のパラメータと拘束条件

パラメータ	p1	寸法値 (単位: m)
	p2	寸法値 (単位: m)
	p3	寸法値 (単位: m)
	p4	寸法値 (単位: m)
	p5	寸法値 (単位: m)
	p6	角度 (単位: 度)
制約条件	p1とp2、p4とp5は垂直であること	
	p2とp4は平行であること	
	p6は0度より大きく90度未満であること	
スケッチ	p1からp4、p6で構成される平面とする	
操作履歴	p5による押し出しとする	

本手法では、MMS や地上型レーザスキャナ等を用いて取得した現況橋梁の点群データをパラメトリックモデルのテンプレートに重ね合わせ、最もフィットイ

ングする寸法値を組合せ最適化アルゴリズムによって導出する。処理の流れを図-37に示す。提案手法における入力データは部位毎に分割済みの点群データとし、出力は各種パラメータが決定済みのパラメトリックモデルとする。なお、計測成果の点群データを部位毎に分割する処理は、点群編集アプリケーションを用いて手動で行うか、後述の機械学習による自動識別手法を想定する。組合せ最適化には、遺伝的アルゴリズム（Genetic Algorithms、以下 GA という）を使用する。GA は、生物が交叉、突然変異、淘汰を繰り返して環境に適合する進化の過程を模したアルゴリズムである。構築手法は、テンプレートの選択、初期集団の生成、選択・交叉、突然変異、個体評価で構成される。

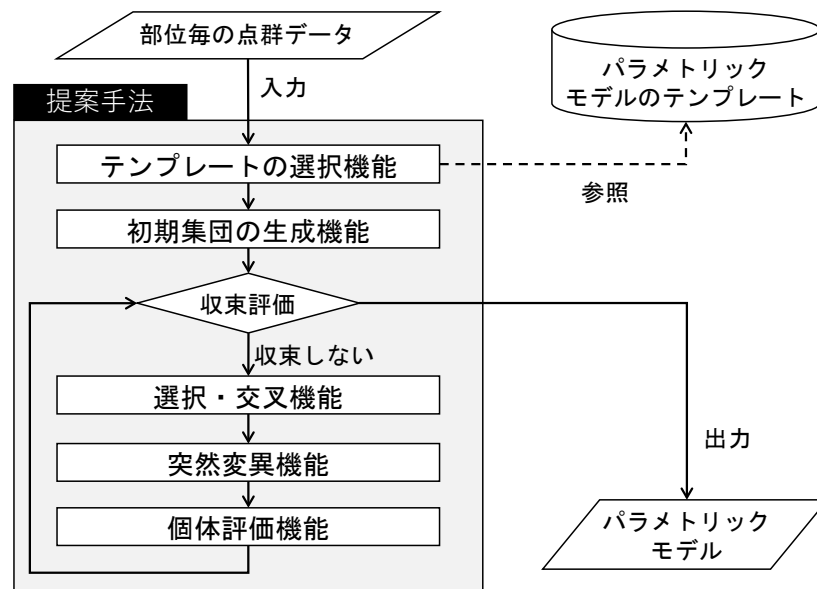


図-37 処理の流れ

## (2) テンプレートの選択

テンプレートの選択では、入力データの部位に応じて、あらかじめ定義したパラメトリックモデルのテンプレートから一つを選択する。

## (3) 初期集団の生成

初期集団の生成では、テンプレートの各パラメータの最適解を GA で導出するため、パラメータを遺伝子配列に置換して初期集団を生成する。まず、パラメー

タの探索範囲と探索ステップ数を設定し、各パラメータを2進数で表現するために必要なビット数を整理する。次に、設定した探索範囲内からランダムな数値を算出し、仮モデルを一つ生成する。ただし、入力点群データを内包するバウンディングボックスと生成した仮モデルのバウンディングボックスが許容値外の場合は、あらためて仮モデルを生成し直す。これを許容値内になるまで繰り返し、生成された仮モデルの重心を点群データの重心に一致するように平行移動させ、座標系を統一する。最後に、平行移動量 (X、Y、Z) と回転角 (roll、pitch、yaw) の探索範囲と探索ステップ数を設定し、テンプレートの各パラメータと併せて2進数の遺伝子配列を定義する。

#### (4) 選択・交叉

選択・交叉では、図-38 に示すように、多様な個体生成を実現するため、任意世代の全個体を対象に、エリート生存、上位個体同士の交叉、エリートとその他個体の交叉、ランキング方式による交叉、ランダム方式による交叉を適用する。エリート生存は、適合度の高い上位個体をそのまま次世代に残すことである。上位個体同士の交叉は、適合度の高い上位個体から無作為に2個体を選択して交叉することである。エリートとその他個体の交叉は、適合度が最も高い1個体とその他いずれかの1個体で交叉することである。ランキング方式による交叉は、適合度が高いほど選択確率を高くなるように設定し、任意の2個体を選択して交叉することである。ランダム方式による交叉は、適合度に関係なく、ランダムに2個体を選択して交叉することである。

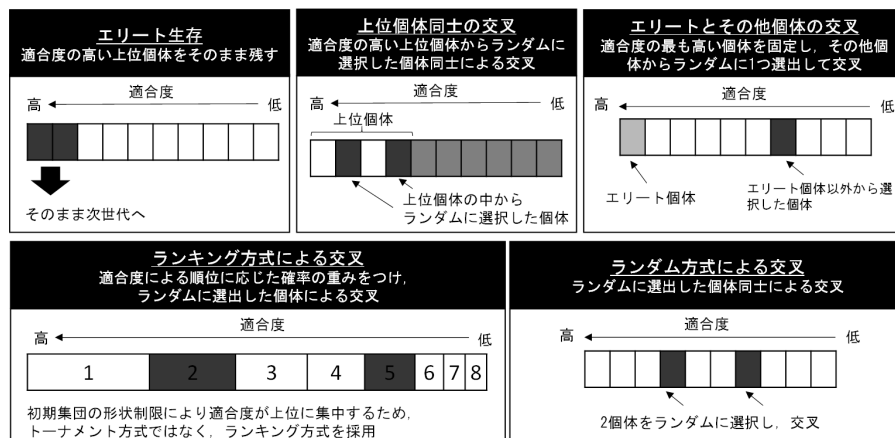
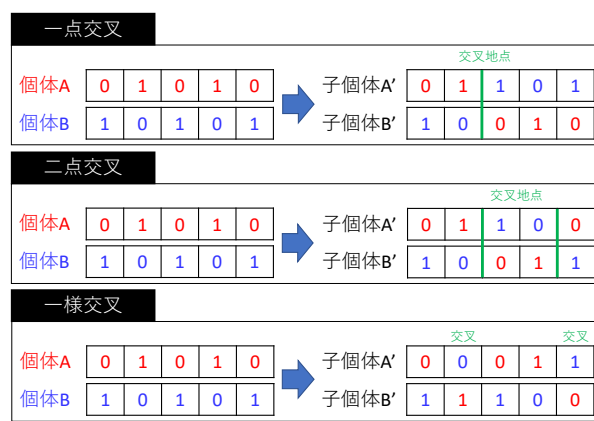


図-38 選択・交叉

交叉方法は、**図-39**に示すとおり、GAの代表的手法として、一点交叉、二点交叉、一様交叉があげられる。一点交叉は、遺伝子配列からランダムに設定した一地点を基準に、他の親個体の遺伝子配列と交換する。二点交叉は、これを二地点で行う。一様交叉は、全ての遺伝子に対して、一定確率で親個体の遺伝子と交換する。

上述の交叉対象個体の選択と交叉方法は、事前のパラメトリック解析により最適なものを決定する。



**図-39 交叉方法**

**(5) 突然変異**

突然変異では、局所解への収束を避けるため、一定確率で遺伝子の値を変換する。構築手法では、遺伝子は2進数表現のため、一定確率で0と1の数値を変換する。

**(6) 個体評価**

個体評価では、GAを用いて生成したパラメトリックモデルと点群データの適合度を算出する。適合度は、**図-40**に示すとおり、まず点群データと生成したパラメトリックモデルを内包する共通の3次元グリッド(立方体)を作成する。次に、全てのグリッドに対して、グリッド内の点または面の有無を確認する。この処理は、点群データと生成したパラメトリックモデルでそれぞれ個別に実行する。そして、算出結果の適合率、再現率とF値を算出する。本手法における適合率は、

パラメトリックモデルから算出した「面が存在するグリッド」の内、点群データから算出した「点が存在するグリッド」がどの程度含まれているかを示す。再現率は、点群データから算出した「点が存在するグリッド」の内、パラメトリックモデルから算出した「面が存在するグリッド」がどの程度含まれているかを示す。F 値は、適合率と再現率を組み合わせた調和平均である。すなわち、F 値が高いと、点群データとパラメトリックモデルが一致していると判断できる。したがって、提案手法では、F 値を生成個体の適合度とみなす。

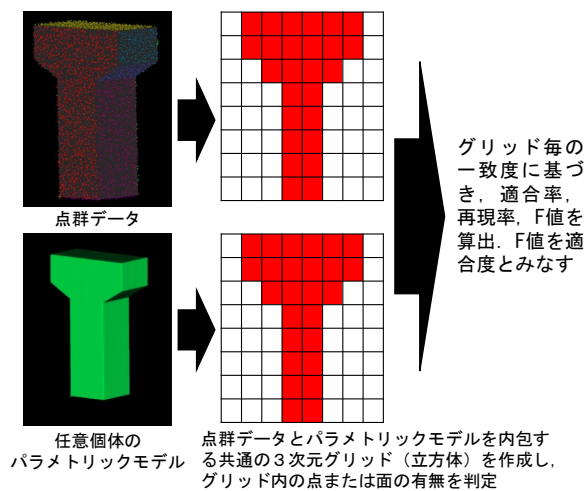


図-40 適合度の算出方法

### 3-4-2 組合せ最適化アルゴリズムによる3次元モデル構築例

組合せ最適化アルゴリズムによる3次元モデル構築例を3つ示す。具体的には、疑似点群データを用いた3次元モデルの構築、未計測箇所を含む疑似点群データを用いた3次元モデルの構築、及び、計測点群データを用いた3次元モデルの構築である。各種パラメータは、パラメトリック解析により導出した表-2に示す最適解を使用する。対象部位(図-41)は、橋梁上部工が床版1種類(箱桁)、下部工が橋脚4種類(T型橋脚、一柱式橋脚、二柱式橋脚、ラーメン橋脚)とする。構築に使用するテンプレートの各パラメータを表-3に示す。

表-2 パラメータと設定値

提案手法のパラメータ	設定値
初期集団の生成条件	バウンディングボックスの大きさが許容値 0.10m以内
初期集団の生成個体数	100
収束条件（最大世代数）	100
収束条件（適合度）	適合度の最大値が50世代変わらない場合は 収束と判断
突然変異確率	25%
交叉方法	一様交叉
エリート生存個体数	適合度の上位2個体
エリート個体同士の交叉時の対象条件	適合度の高い順から10個体
エリート個体同士の交叉個体数	8個体
エリートとその他個体の交叉個体数	40個体
ランキング方式の交叉個体数	30個体
ランダム方式による交叉個体数	20個体
適合度算出のグリッドサイズ	0.10m

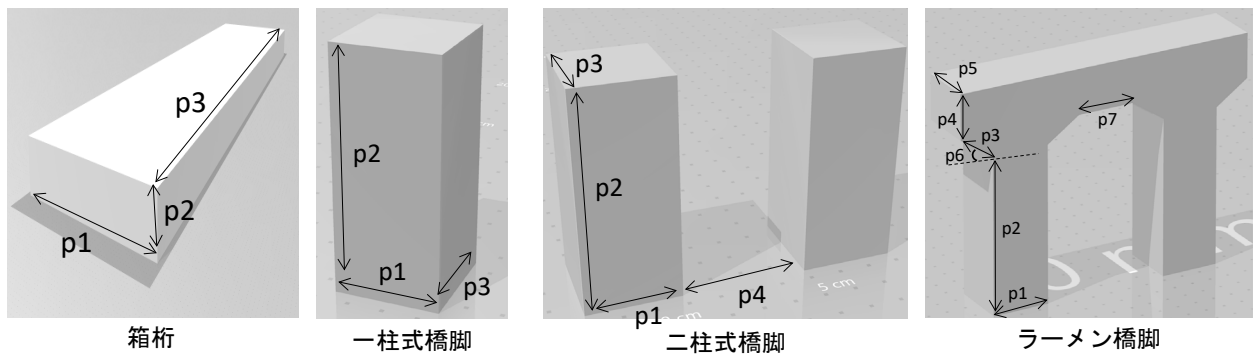


図-41 モデルのテンプレート

表-3 テンプレートのパラメータ

部位	パラメータ	探索ステップ	探索範囲	パターン	必要ビット
箱桁	p1	0.10m	2.00~8.40m	64	6bit
	p2	0.10m	0.10~1.60m	16	4bit
	p3	0.10m	18.00~24.40m	64	6bit
T型橋脚	p1	0.25m	0.50~4.50m	16	4bit
	p2	0.25m	2.00~10.00m	32	5bit
	p3	0.25m	2.00~10.00m	32	5bit
	p4	0.25m	0.50~4.50m	16	4bit
	p5	0.25m	1.00~5.00m	16	4bit
	p6	5.00 度	10.00~90.00 度	16	4bit
一柱式橋脚	p1	0.50m	0.50~8.00m	16	4bit
	p2	0.50m	3.00~18.00m	32	5bit
	p3	0.50m	0.50~4.00m	8	3bit
二柱式橋脚	p1	0.50m	0.50~8.00m	16	4bit
	p2	0.50m	3.00~18.00m	32	5bit
	p3	0.50m	0.50~4.00m	8	3bit
	p4	0.50m	0.50~8.00m	16	4bit
ラーメン橋脚	p1	0.50m	0.50~8.00m	16	4bit
	p2	0.50m	3.00~18.00m	32	5bit
	p3	0.50m	0.50~8.00m	16	4bit
	p4	0.50m	0.50~8.00m	16	4bit
	p5	0.50m	0.50~4.00m	8	3bit
	p6	10.00 度	10.00~80.00 度	8	3bit
	P7	0.50m	0.50~8.00m	16	4bit

## (1) 疑似点群データを用いた構築例

### 1) 構築方法

まず、任意のパラメータを与えて T 型橋脚とラーメン橋脚を生成する。次に、その各面上に点を生成する。このとき、一定範囲の乱数を発生させ、点の座標値を調整する。これにより、疑似的に対象部位を計測した点群データを生成する。そして、疑似点群データとモデルテンプレートから任意に設定したパラメータを推定できることを確認する。ここでは、直近 5 年程度の計測成果に対して本技術が適用できることを目指して、正解平面から  $\pm 0.025\text{m}$  となるように疑似点群を生成した。

### 2) 構築結果

提案手法の適用結果を図-42 に示す。図中の最終適合度とは、構築手法を用いて生成した個体内、最も高い適合度を示す。また、正解適合度とは、疑似点群データ生成時に設定したパラメータに基づき算出した適合度を示す。

T 型橋脚は 5 世代目、ラーメン橋脚は 34 世代目で最大適合度の個体を生成できた。そして、この個体のパラメータは、疑似点群データ生成時のパラメータと一致した。適合度は、ともに 0.98 であった。以上より、本手法を用いて、点群データから 3 次元モデルを構築できた。

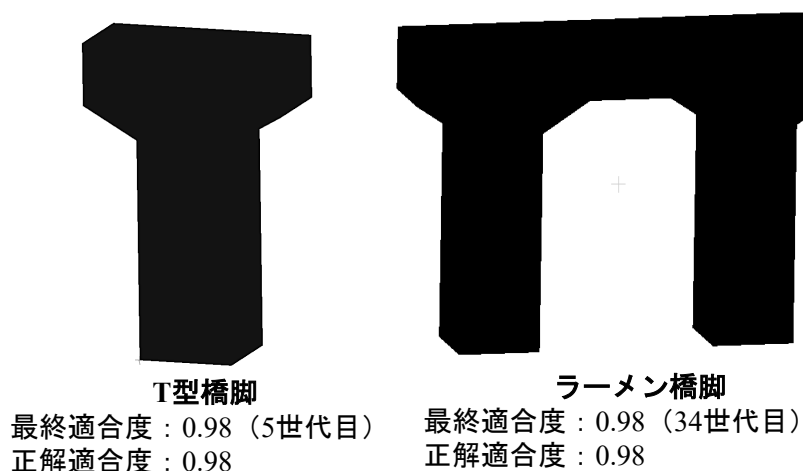


図-42 疑似点群データから 3 次元モデルの構築結果



## (2) 未計測箇所を含む点群データを用いた構築例

### 1) 構築方法

まず、T型橋脚、一柱式橋脚、二柱式橋脚とラーメン橋脚の疑似点群データを生成する。次に、計測不可箇所の上側と下側の点を削除する。最後に、一方向からの計測を想定し、**図-43**に示すとおり片側の点を全て削除する。これらの疑似点群データを用いて3次元モデルを構築する。

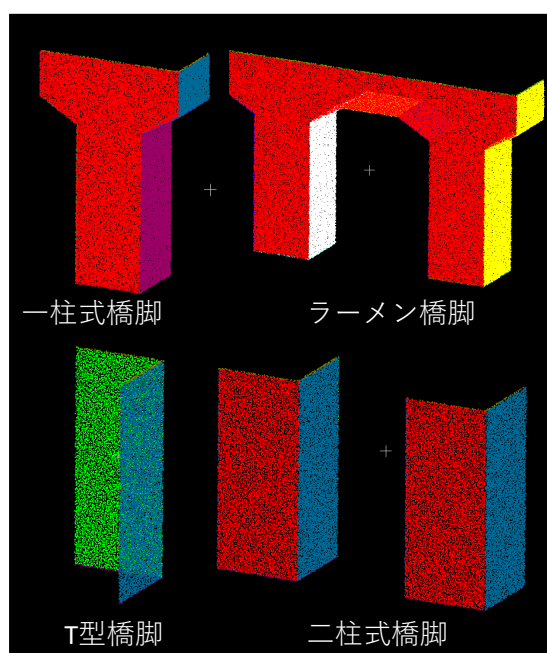


図-43 一方向からの計測を想定した疑似点群データ

## 2) 構築結果

構築結果を図-44 に示す。一柱式橋脚と二柱式橋脚は、疑似点群データ生成時のパラメータを推定できた。一方、T型橋脚とラーメン橋脚は、いずれもパラメータ p5 のみ、疑似点群生成時のパラメータと比べて 0.50m 短い結果となった。また、個体の適合度は、両方とも正解適合度よりも高くなった。この要因としては、適合度の算出基準となる 3次元グリッドの区切り位置と、疑似点群データ生成時の乱数による点の揺れに起因すると考えられる。この結果のように、疑似点群データの分布次第では、正解の寸法値よりも小さいモデルの方が、F 値が高くなるケースが見られた。

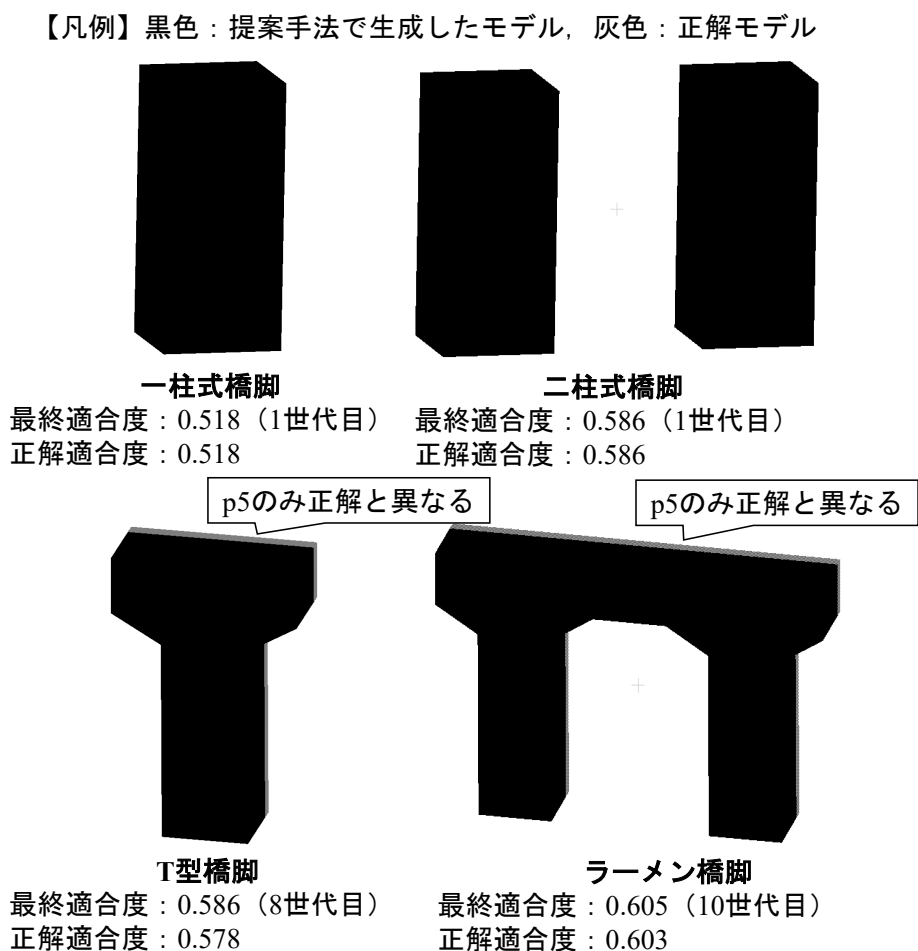


図-44 一方向からの計測を想定した疑似点群データを用いた 3次元モデルの構築結果

### (3) 計測点群データを用いた構築例

#### 1) 構築方法

本例では、地上型レーザスキャナを用いて取得した橋梁上部工の箱桁と、MMSを用いて取得した高架道路橋の橋脚の点群データ（図-45）を対象に3次元モデルを構築する。箱桁は、大阪府泉南市信達童子畑地区にある童子畑橋とする。高架道路橋の橋脚は、阪神高速11号池田線高架橋下とする。隣接道路を走行しながらMMSで計測したため、一方向からの計測に限定され、未計測箇所が多い状態である。正解パラメータは、点群データから目視判読により手動で設定した。

#### 2) 構築結果

実験結果を図-46に示す。箱桁では、構築手法により、第5世代目に適合度0.68の個体が生成された。これは、正解パラメータと同一であった。そのため、橋台及び路面との接触箇所の点群が欠損している状態であっても、提案手法により正確な箱桁のパラメトリックモデルが生成できることを確認した。橋脚は、構築手法により、第12世代目に適合度0.55の個体が生成された。推定結果と正解パラメータを確認すると、 $p_2$ が0.50m短く、 $p_4$ が0.25m短く、 $p_6$ が5度大きかった。対象構造物の現況と点群データを分析した結果、構造物を上から確認した図-47に示すとおり、地面と設置する直方体部と上部工を支える部位の大きさが約0.50m異なっていた。この部位は、T型橋脚のモデルパラメータ $p_5$ に該当する。

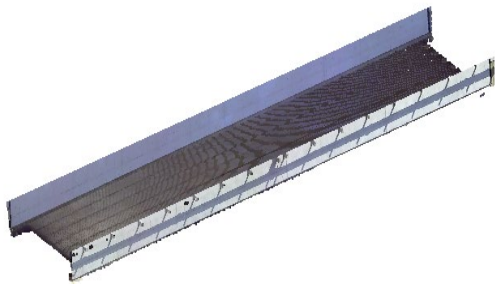
以上より、構築手法により、地上型レーザスキャナやMMS等の計測機器を用いて取得した点群データ及び未計測箇所を含む点群データに対しても適用可能であった。



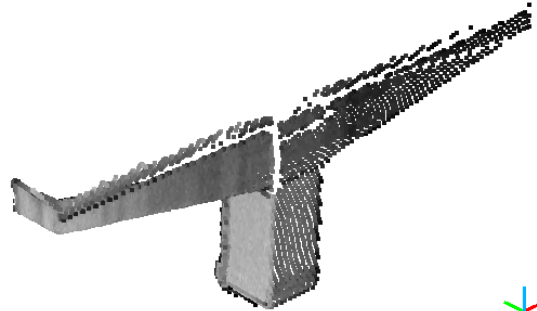
橋梁の現場写真



高架橋下の現場写真



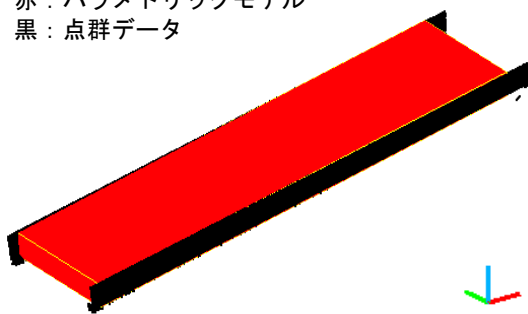
箱桁の点群データ



橋脚の点群データ

図-45 計測環境と点群データ

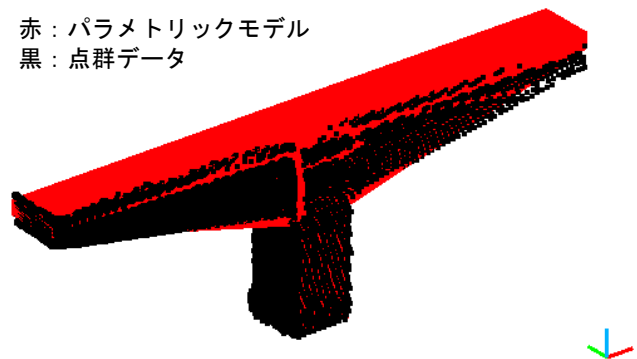
赤：パラメトリックモデル  
黒：点群データ



箱桁

最終適合度：0.68（5世代目）  
正解適合度：0.68

赤：パラメトリックモデル  
黒：点群データ



橋脚

最終適合度：0.55（12世代目）  
正解適合度：0.57

図-46 計測点群データを用いた3次元モデルの構築結果

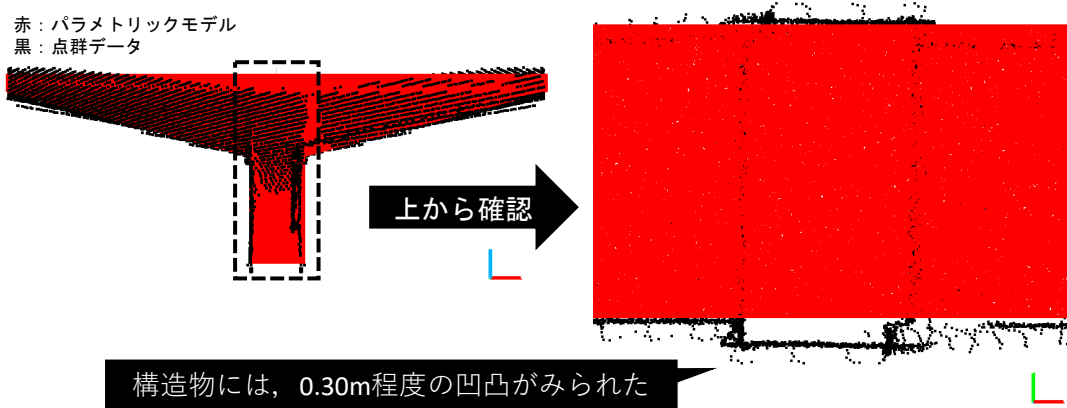


図-47 構造物の凹凸

### 3-5 ISO10303 に準拠したデータ構造

パラメトリックモデルのデータ構造は、CADにおけるパラメトリックモデリングとの連携を図るため、図-48に示すとおり、ISO10303 (STEP: Standard for the Exchange of Product model data) に準拠する。具体的には、2次元のスケッチ平面上で対象構造物の横断または縦断形状を作成し、それを押し出し操作によって3次元の幾何形状を構築する。幾何フィーチャは、互いに平行や直角等の拘束条件を保持する。これにより、任意の寸法値を変更すると、位相を保持した状態でモデルが再生成される。

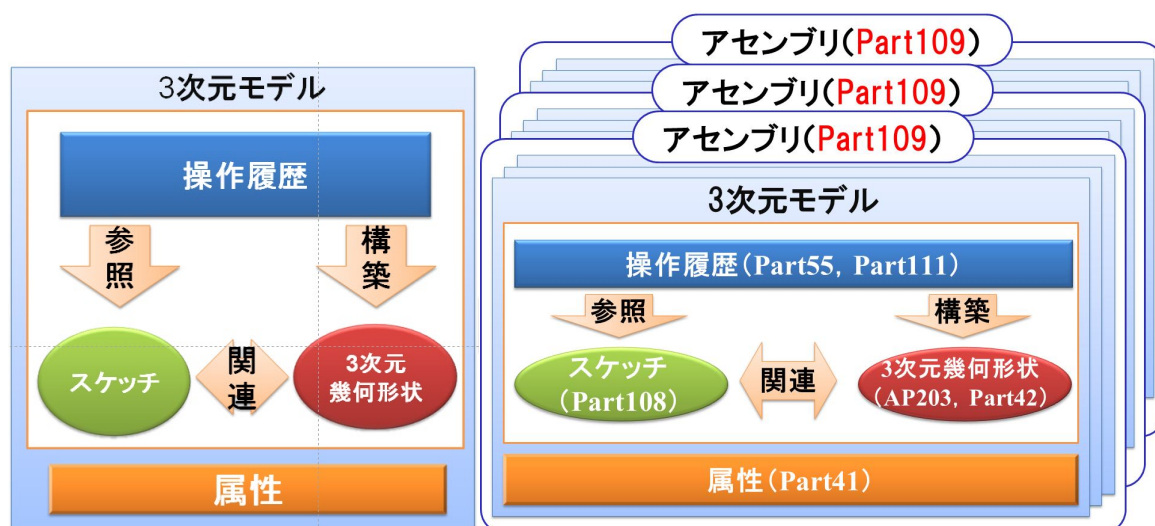


図-48 ISO10303 (STEP) に準拠したデータ構造

#### 3-5-1 適用する ISO10303 の仕様

本指針では、ISO10303 に準拠した操作履歴、スケッチ、アセンブリおよび属性のデータ構造を定義する。これらに関連する ISO10303 の仕様はそれぞれ次のとおりである。

なお、データ構造では、関西大学カイザー・プロジェクト「時間項を考慮した3次元CADエンジンの開発」及び国土交通省建設技術研究開発助成「汎用3次元CADエンジンの調査と設計に関する技術開発」の成果を活用した。

## (1) Part55 (操作履歴)

### 1) 概要

ISO10303-55 (以下、Part55 という) は、“Procedural and hybrid representation (手続き型及びハイブリッド表現)” として規格化されている。Part55 は、モデルを構築する操作に関して定義されたモデル表現のための汎用的な方法を規定している。そして、手続き型とハイブリッド型に関する CAD 形状表現の交換を対象とする。ハイブリッド表現は手続きだけでなく明示的な要素 (幾何形状モデル) を含む。構築操作自身は Part111 で定義される。手続き型モデルは、構築操作の引数として使われるパラメータ値を変更することによって容易に編集できるという利点を有する。

Part55 と Part111 の考え方では、3 次元データを形成するための操作履歴は図-49 に示すように表現される。

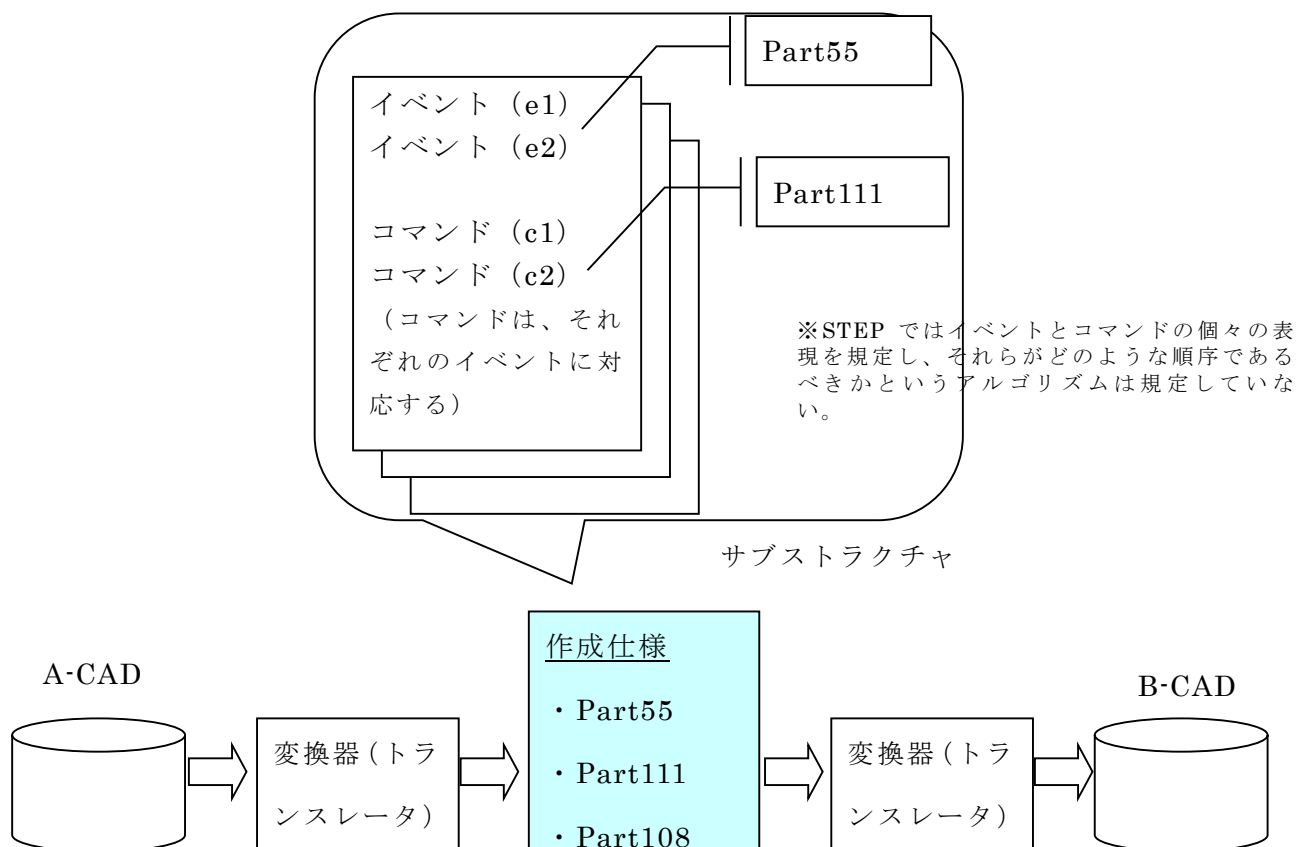


図-49 ISO10303 における操作履歴の保持

Part55 の大分類は次のスキーマである。

- Procedural model schema (手続き型モデル)

procedural\_model\_schema は、手続き型およびハイブリッド型モデルの表現、そして設計根拠の記録のための基本的な方法を定義する。

- Procedural shape model schema (手続き型形状モデル)

procedural\_shape\_model\_schema は、上記スキーマの特殊なスキーマであり、幾何モデルに特化して利用する定義である。

## 2) 適用範囲

Part55 は、手続き型あるいは構築履歴型のモデル表現を規定している。これらは、モデルを作成するために利用される構築操作の順序を定義している。操作自体の表現は Part55 では規定していない。なお、Part108 で定義される機能は、Part111 で定義される機能と密接に関連付いている。

Part55 のスコープは以下のものである。

- あらゆる種類の明示的な表現、あるいは、モデルを生成するための構築操作の順序の定義
- 構築順序の階層的な構造化 (サブストラクチャ)
- ハイブリッドモデルの表現のための構築順序における明示的に定義される要素の組み込み (明示的な幾何要素を構築順序に関連づけるための仕様)

## 3) Procedural model schema (手続き型モデル)

procedural\_representation\_schema は構築操作に関する表現あるいはモデリングを規定する。これは操作の結果として明示的に生成される要素の表現あるいはモデリングを規定する。本スキーマは以下の表現方法を規定する。

- あらゆる種類のモデルの生成あるいは表現に関する構築操作の順序 (操作履歴) の定義
- 構築順序 (操作履歴) の階層的な構造 (サブストラクチャ)
- ハイブリッドモデルの表現のための構築順序 (操作履歴) における明示的に定義される要素の組み込み (明示的な幾何要素を構築順序に関連づけるための仕



様)

- “representation\_item” によって表現されるインスタンスのための構築操作（操作履歴）を表現する “representation\_item” の利用
- 手続き型モデルと明示的な ‘current result’ モデル（その時点での構築結果）の関連付けによる表現の定義。後者は、前者によって定義されるモデルのパラメトリックを表現する例としての役割を持つ。つまり、その時点での構築結果を操作履歴によって再実行した際にどのように形状表現されるかを表す。
- 手続き型モデルにおける設計意図情報の関連付け
- 手続き型モデルにおいて、ディスプレイに表示されたモデルからインタラクティブな操作によって選択される明示的な要素の識別を表す。つまり、CAD システムの画面上に表示される要素をマウスなどで選択される要素である。
- 手続き型モデルにおいてモデル簡略化のために抑制する構築操作（操作履歴）の識別

#### 4) Procedural shape model schema（手続き型形状モデル）

procedural\_shape\_model\_schema は構築操作に関する形状の表現あるいはモデリングを主題とする。本スキーマは以下の表現方法を規定する。

- 構築操作の順序に関する形状表現の仕様。
- ISO10303-42 で規定される形状モデリングのエンティティの使用。それらのエンティティ型のインスタンスにおける構築操作を表現するためのもの。

Procedural shape models は、procedural\_shape\_representation エンティティを有し、procedural\_representation のサブタイプである。このスキーマでは、操作履歴は ISO10303-42 で定義される geometric\_representation\_item と topological\_representation\_item のインスタンスを生成する。

## (2) Part111 (操作履歴)

### 1) 概要

ISO10303-111 (以下、Part111 という) は、“Elements for the procedural modeling of solid shapes (ソリッド形状の手続き型モデリングの要素)” として規格化されている。これは、CAD システムのソリッドモデリング機能によって作成される複雑な形状要素を表現するための方法を定義している。その要素は、操作履歴によって表現される製品のソリッド形状の交換を行うために定義される。このような製品形状モデルの表現は、手続き型、ヒストリベース、あるいは構築履歴モデルと呼ばれる。規定されているスキーマは、“solid\_shape\_element\_schema” である。これは、CAD システムにおいてソリッド形状モデルに組み込むことができる複雑な幾何要素を定義する。(Part111 を構成するスキーマは、“solid\_shape\_element\_schema” のみである。)

### 2) 適用範囲

Part111 の適用範囲は以下のとおりである。

- 結合や面取り操作によるソリッド形状の表現
- オフセット、厚み付け、シェル化によるソリッド形状の表現
- 穴、くぼみ、隙間、溝により生成されるソリッド形状の表現
- 前述の特徴を有する円と長方形のパターンにより生成されるソリッド形状の表現
- ISO10303-42 で定義された押し出しと回転のソリッド形状の表現の一般化

### 3) Solid shape element schema

Part111 を構成するスキーマは、“Solid shape element” スキーマのみである。

CAD システムのユーザが利用できる基本的な操作は以下のものである。

- a) 2次元スケッチを基礎とした押し出しと回転の操作
- b) 簡単な形状を組み合わせて複雑な形状を生成するためのブール演算
- c) 既存のソリッドモデルにおいて部分的な修正を行う、その CAD システム固有の操作

このうち、aタイプとbタイプは手続き型モデルのキャプチャとCADシステム間の転送においてPart55が規定している。cタイプはPart111が規定している。

### (3) Part108 (スケッチ・拘束)

#### 1) 概要

ISO10303-108 (以下、Part108 と言う) は、“Parameterization and constraints for explicit geometric product models (明示的な幾何学的なプロダクトモデルのためのパラメトリゼーションと拘束)” として規格化されている。Part108 は、代数拘束と幾何拘束の 2 通りの拘束とアプリケーションとしてのスケッチを定義している。Part108 は、当時不可能であった、可変なモデルの定義を目的としている。

#### 2) 適用範囲

Part108 の適用範囲は以下のとおりである。

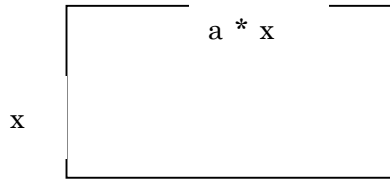
- 可変量の連携によるモデルのパラメトリゼーション
- パラメータ間の数学的拘束
- モデルの構成要素や属性間の関係として表現される拘束
- 製品、計画、過程、組織の可変モデルを生成するための表現レベルでのパラメータの連携と拘束
- 製品形状モデルに一般的に用いられる幾何拘束のための特定の表現
- 2、3 次元形状モデル、特に 2 次元幾何スケッチパラメトリゼーションと拘束の応用
- 不完全に拘束されたとみなされるモデルの表現

#### 3) Parameterization schema

ISO10303 で定義されたモデルに対して、可変のパラメータ (変数) をデータ交換する場合に利用する。

以下の 2 つのエンティティが重要。

- `bound_model_parameter` (関連のある変数)  
モデルの属性と関連のある変数
- `unbound_model_parameter` (関連のない変数)  
モデルの属性と関連のない変数



この例の場合、モデルの属性の辺の長さ  $x$  が `bound_model_parameter` で、 $x$  の  $a$  倍というように辺の長さを表す場合の  $a$  が `unbound_model_parameter` となる。

#### 4) Explicit constraint schema

ユーザによって数式などで定式化された拘束をデータ交換する場合に利用する。大きく分けて2つのエンティティがある。

- `defined_constraint` (明確な数式的拘束)
  - `defined_constraint` の下位のエンティティ
  - `equal_parameter_constraint` 等価な拘束 (例:  $x=y$ )
- `free_form_constraint` (自由な数式的拘束)
  - `free_form_constraint` の下位のエンティティ
  - `free_form_assignment` 割り当てを表す拘束
  - `free_form_relation` 関係を表す拘束 (例:  $x<y$ )

#### 5) Variational representation schema

データ交換時に受信側が送り手の意図を満たす範囲での変更を可能にするため、拘束情報やパラメータを関連付けるスキーマである。可変モデリングを新たに実現するためのベースとなるスキーマと考えられる。

#### 6) Sketch schema

##### (A) 概要

- スケッチ図面をデータ交換する場合に利用する。
- スケッチは、代数拘束と幾何拘束の2通りの拘束のアプリケーションを有する。
- スケッチには、適当な平面上のスケッチと3次元座標上のスケッチの2通りある。

(B) エンティティ

- implicit point on plane → 平面上の点のスケッチ
- implicit planar intersection point → 平面上の交点のスケッチ
- implicit planar projection point → 平面上の投影点のスケッチ
- implicit planar curve → 平面上の曲線のスケッチ
- implicit intersection curve → 平面上の交曲線のスケッチ
- implicit projected curve → 平面上の投影曲線のスケッチ
- implicit model intersection curve
- implicit silhouette curve
- neutral sketch representation → 適当な平面上のスケッチ
- positioned sketch → 3次元座標上のスケッチ
- repositioned neutral sketch
- implicit explicit positioned sketch relationship
- subsketch
- rigid subsketch

#### (4) Part109 (アセンブリ)

##### 1) 概要

Part109 は、Part108 で確定された部品に対して、部品間の関係（配置）を定義している。Part109 は、Part108 をアセンブリに特化したものである。

##### 2) 適用範囲

Part109 の適用範囲は以下のとおりである。

- 形状関係とその表現の結合
- アセンブリフィーチャの関係とその表現の結合
- アセンブリの幾何学的な制約、`geometric_representation_item` レベルにおいてのアセンブリフィーチャ関係の幾何学的情報の表現
- アセンブリモデルにおける役割を果たす固定された構成要素の表現

##### 3) Assembly feature relationship schema

`assembly_feature_relationship_schema` は、アセンブリフィーチャの関連性について定義したスキーマである。

##### 4) Assembly constraint schema

`assembly_constraint_schema` は、アセンブリ拘束について定義したスキーマである。

## (5) Part41 (属性)

### 1) 概要

Part41 (製品の記述および支援の基礎) は統合共通リソースであり、他の統合共通リソースにおいて用いられる表現要素の中で、単位系や年号など共通に用いられるものの表現を集めた分冊である。ユーザが独自に設定できる属性が定義されており、その他の Part にドメイン毎の属性が定義されている。

### 2) 適用範囲

- “generic product description resource” では、製品の同定、製品の分類、バージョン仕様、それらの関係、製品の可能な代用が範囲となる。
- “generic management resources” では、管理データを製品データに関連付けさせるために用いる構造のための資源を提供する。
- “support resources” では、管理データ、物理的量、単位、基本データタイプのための資源を提供する。

### 3) External reference schema

Part41 において、external\_reference\_schema は、オブジェクトに対して自由に文字列情報を付与することができるスキーマである。このスキーマの考えを利用して、提案する交換仕様における属性の定義を実現できると考える。



### 3-5-2 操作履歴

本項では、操作履歴の仕様として、Part55 と Part111 のエンティティを列挙し、その説明、定義および条件を示す。各エンティティ名に記載される括弧内の番号は、ISO 規格書（ISO10303-55 または ISO10303-111）の該当項を表す。

#### (1) Procedural model スキーマ (Part55)

##### 1) 明示的な手続き型表現

エンティティ名	explicit_procedural_representation_relationship (4.3.1)
<p>&lt;説明&gt;            手続き型あるいはハイブリッド表現と、これに一致する明示的な表現（current result：その時点での構築結果）を関連づける。</p>	
<p>&lt;EXPRESS による定義&gt;            ENTITY explicit_procedural_representation_relationship                SUBTYPE OF (representation_relationship);                SELF¥representation_relationship.rep_1 : procedural_representation;            WHERE                WR1 : (NOT ('PROCEDURAL_MODEL_SCHEMA.PROCEDURAL_REPRESENTATION'                            IN TYPEOF(SELF¥representation_relationship.rep_2))) AND                (NOT ('VARIATIONAL_REPRESENTATION_SCHEMA.VARIATIONAL_REPRESENTATION'                            IN TYPEOF(SELF¥representation_relationship.rep_2)));                WR2 : SELF¥representation_relationship.rep_1.context_of_items ::=                        SELF¥representation_relationship.rep_2.context_of_items;            END_ENTITY;</p>	
<p>&lt;条件&gt;            ・属性 “SELF¥representation_relationship.rep_2” によって参照される表現は、“procedural_representation” あるいは “variational_representation” のインスタンスとならない。            ・“explicit_procedural_representation_relationship” のインスタンスによって関連づけられる 2 つの表現は表現コンテキストを共有する。</p>	

## 2) 明示的な手続き型表現の項目

エンティティ名	explicit_procedural_representation_item_relationship (4.3.2)
<p>&lt;説明&gt;          手続きの要素と明示的要素（幾何要素）を結びつけて個々の集まりを作り上げる。このエンティティは、Part43（表現構造）で規定される representation_item のサブタイプである。</p>	
<p>&lt;EXPRESS による定義&gt;          ENTITY explicit_procedural_representation_item_relationship          SUBTYPE OF (representation_item_relationship);          SELF¥representation_item_relationship.relateing_representation_item :          procedural_representation_sequence;          WHERE          WR1 : NOT          ('PROCEDURAL_MODEL_SCHEMA.PROCEDURAL_REPRESENTATION_SEQUENCE'          IN TYPEOF(          SELF¥representation_item_relationship.related_representation_item));          WR2 : SIZEOF(QUERY(q &lt;* using_representations(          SELF¥representation_item_relationship.related_representation_item)            item_in_context(          SELF¥representation_item_relationship.relateing_representation_item,          q.context_of_items))) &gt; 0;          END_ENTITY;</p>	
<p>&lt;条件&gt;          ・属性“SELF¥representation_item_relationship.related_representation”によって参照される表現は、“procedural_representation_sequence”のインスタンスと          ならない。          ・2つの関連づけられた“representation_item”のインスタンスは、少なくとも1          つの共通の“representation_context”を持つ。</p>	

## 3) 手続き型表現

エンティティ名	procedural_representation (4.3.3)
<p>&lt;説明&gt;          procedural_representation_sequence のインスタンスに関するモデルを定義する。操作順序（sequence）はモデル全体あるいは一部を表現する。</p>	
<p>&lt;EXPRESS による定義&gt;          ENTITY procedural_representation          SUBTYPE OF (representation);          SELF¥representation.items :          SET[1:?] OF procedural_representation_sequence;          END_ENTITY;</p>	

### (A) 手続き型表現の順序

エンティティ名	procedural_representation_sequence (4.3.4)
<説明> 一部あるいは完全なモデルを構築するための操作のリストを表現する。	
<EXPRESSによる定義> ENTITY procedural_representation_sequence SUBTYPE OF (representation_item); elements : LIST[1:?] OF representation_item; suppressed_items : SET[0:?] OF representation_item; rationale : text; WHERE WR1: SIZEOF(QUERY(q <* suppressed_items   NOT (q IN elements))) = 0; END_ENTITY;	
<条件> 操作順序に存在しない“suppressed_items”（削除されたアイテム）において生じるアイテムはない。	

### (B) ユーザ選択による要素

エンティティ名	user_selected_elements (4.3.5)
<説明> 送信側システムの画面でピックされた 1 つ以上の明示的要素の参照を表現する（つまり、ユーザが画面を見てピックした要素を表現する）。	
<EXPRESSによる定義> ENTITY user_selected_elements SUBTYPE OF (representation_item); picked_items : SET[1:?] OF representation_item; END_ENTITY;	

### (C) 間接的に選択された要素

エンティティ名	indirectly_selected_elements (4.3.6)
<説明> 送信された構築履歴から生成される明示的なモデル（3次元幾何形状）に相当する要素に一致するインスタンスを表現する。	
<EXPRESSによる定義> ENTITY indirectly_selected_elements SUBTYPE OF (user_selected_elements); indirectly_picked_items : SET[1:?] OF representation_item; END_ENTITY;	

(2) Procedural shape model スキーマ (Part55)

(A) 明示的な手続き型による形状表現の関係

エンティティ名	explicit_procedural_shape_representation_relationship (5.4.1)
<p>&lt;説明&gt;                  手続き型あるいはハイブリッド表現と、これに一致する明示的な current result (その時点での構築結果) を関連づける。</p>	
<p>&lt; EXPRESS による定義 &gt;                  ENTITY explicit_procedural_shape_representation_relationship                  SUBTYPE OF (explicit_procedural_representation_relationship);                  SELF¥representation_relationship.rep_1 :                  procedural_shape_representation;                  SELF¥representation_relationship.rep_2 : shape_representation;                  END_ENTITY;</p>	

(B) 明示的な手続き型による幾何要素表現の項目の関係

エンティティ名	explicit_procedural_geometric_representation_item_relationship (5.4.2)
<p>&lt;説明&gt;                  幾何表現項目の 2 つのインスタンスである同じ形状を表現する手続きと明示的表現を関連づける。</p>	
<p>&lt; EXPRESS による定義 &gt;                  ENTITY explicit_procedural_geometric_representation_item_relationship                  SUBTYPE OF (explicit_procedural_representation_item_relationship);                  SELF¥representation_item_relationship.relying_representation_item :                  procedural_shape_representation_sequence;                  SELF¥representation_item_relationship.related_representation_item :                  geometric_representation_item;                  WHERE                  WR1 : NOT (                  'PROCEDURAL_SHAPE_MODEL_SCHEMA.PROCEDURAL_SHAPE_ REPR                  REPRESENTATION_ SEQUENCE'                  IN TYPEOF(                  SELF¥representation_item_relationship.related_representation_item));                  END_ENTITY;</p>	
<p>&lt;条件&gt;                  “representation_item” のインスタンスは、属性 “SELF¥representation_item_ relationship.related_representation” によって参照されるが、type “procedural_ shape_representation_sequence” の一部とはならない。</p>	

### (C) 手続き型形状表現

エンティティ名	procedural_shape_representation (5.4.3)
<説明> procedural_shape_representation_sequence のインスタンスに関するモデルを定義する。	
< EXPRESS による定義 > ENTITY procedural_shape_representation SUBTYPE OF (procedural_representation, shape_representation); SELF%representation.items : SET[1:?] OF procedural_shape_representation_sequence; END_ENTITY;	

### (D) 手続き型形状表現の手順

エンティティ名	procedural_shape_representation_sequence (5.4.4)
<説明> 形状モデルを作成するための構築操作のリストを表現する。構築操作を正しい命令でリスト表現することで、交換された形状モデルは受け取り側のシステムで正しく再構築される。	
< EXPRESS による定義 > ENTITY procedural_shape_representation_sequence SUBTYPE OF (geometric_representation_item, procedural_representation_sequence); WHERE WR1 : SIZEOF(QUERY(q <* SELF%procedural_representation_ sequence.elements   NOT (‘PROCEDURAL_SHAPE_MODEL_SCHEMA.SHAPE_REPRESENTATION_ ITEM’ IN TYPEOF(q)))) = 0; END_ENTITY;	
<条件> 受信側システムで生成される各モデル要素は、SELECT 型の “shape_representation_item” に属するタイプの 1 つである。	

### (E) 手続き型によるソリッド表現の手順

エンティティ名	procedural_solid_representation_sequence (5.4.5)
<説明> ソリッドモデルの構築手順を表現する。	
< EXPRESS による定義 > ENTITY procedural_solid_representation_sequence SUBTYPE OF (procedural_shape_representation_sequence); WHERE WR1 : ‘GEOMETRIC_MODEL_SCHEMA.SOLID_MODEL’ IN TYPEOF(SELF); END_ENTITY;	

<条件>

“procedural\_solid\_representation\_sequence” のインスタンスの型リストは、“solid\_model” を含む。

#### (F) 手続き型によるサーフェス表現の手順

エンティティ名	procedural_surface_representation_sequence (5.4.6)
<説明> サーフェスモデルの構築手順を表現するためのエンティティであり、Part42 で定義される shell_based_surface_model あるいは face_based_surface_model といったサーフェスモデルのインスタンスを含むものである。	
<EXPRESS による定義> ENTITY procedural_surface_representation_sequence SUBTYPE OF (procedural_shape_representation_sequence); WHERE WR1 : (‘GEOMETRIC_MODEL_SCHEMA.FACE_BASED_SURFACE_MODEL’ IN TYPEOF(SELF)) XOR (‘GEOMETRIC_MODEL_SCHEMA.SHELL_BASED_SURFACE_MODEL’ IN TYPEOF(SELF)); END_ENTITY;	
<条件> “procedural_sorface_representation_sequence” のインスタンスの型リストは、“face_based_surface_model” あるいは “shell_based_surface_model” のいずれかを含む。	

#### (G) 手続き型によるワイヤースケルトン表現の手順

エンティティ名	procedural_wireframe_representation_sequence (5.4.7)
<説明> ワイヤースケルトンモデルの構築手順を表現するためのエンティティである。Part42 で定義される shell_based_wireframe_model あるいは edge_based_wireframe_model といったワイヤースケルトンモデルのインスタンスを含むものである。	
<EXPRESS による定義> ENTITY procedural_wireframe_representation_sequence SUBTYPE OF (procedural_shape_representation_sequence); WHERE WR1 : (‘GEOMETRIC_MODEL_SCHEMA.EDGE_BASED_WIREFRAME_MODEL’ IN TYPEOF(SELF)) XOR (‘GEOMETRIC_MODEL_SCHEMA.SHELL_BASED_WIREFRAME_MODEL’ IN TYPEOF(SELF)); END_ENTITY;	

<条件>

“procedural\_wireframe\_representation\_sequence” のインスタンスの型リストは、“edge\_based\_wireframe\_model” あるいは “shell\_based\_wireframe\_model” のいずれかを含む。

### (3) Solid shape element スキーマ (Part111)

#### (A) 編集されたソリッドモデル

エンティティ名	modified_solid (4.4.1)
<説明> 既存ソリッドに新たに構築した要素の結果であるソリッドモデルのサブタイプである。設計意図をテキストで記述できる。	
<EXPRESS による定義> ENTITY modified_solid ABSTRACT SUPERTYPE OF (ONE OF (edge_bleneded_solid, sculptured_solid, shelled_solid, offset_face_solid, modified_solid_with_placed_configuration )) SUBTYPE OF (solid_model); rationale : text; base_solid : base_solid_select; END_ENTITY;	

#### (B) 境界結合されたソリッドモデル

エンティティ名	edge_bleneded_solid (4.4.2)
<説明> 結合モデルの手続き型表現の交換は、オリジナルの結合されていないエッジの明示的な表現を含む交換ファイルを必要とする。これは送信側システムで定義され、受信側システムで識別される。識別メカニズムはPart55のuser_selected_elementsの定義に詳述されている。	
<EXPRESS による定義> ENTITY edge_bleneded_solid ABSTRACT SUPERTYPE OF (track_bleneded_solid ANDOR (ONE OF solid_with_constant_radius_edge_blend, solid_with_variable_radius_edge_blend, solid_with_chamfered_edge))) SUBTYPE OF (modeified_solid); blended_edges : LIST [1:?] OF UNIQUE edge_curve; END_ENTITY;	



(C) 軌道結合されたソリッドモデル

エンティティ名	track_bleded_solid (4.4.3)
<p>&lt;説明&gt;                  連続した開閉路を結合して形作る、結合されていない境界である                  “edge_bleded_solid” インスタンスの1つのタイプである。</p>	
<p>&lt; EXPRESS による定義 &gt;                  ENTITY track_bleded_solid                  ABSTRACT SUPERTYPE OF (track_bleded_solid_with_end_conditions)                  SUBTYPE OF (edge_bleded_solid);                  WHERE                  WR1: check_continuous_edges(SELF¥edge_bleded_solid.blended_edges);                  END_ENTITY;</p>	
<p>&lt;条件&gt;                  “edge_curve” インスタンスにおける各要素の1つの交点は、次の要素の1つの交点と一致する。</p>	

(D) 境界条件を有する軌道結合を有するソリッドモデル

エンティティ名	track_bleded_solid_with_end_condition (4.4.4)
<p>&lt;説明&gt;                  結合が境界の始終点の両方でないソリッドモデルである。</p>	
<p>&lt; EXPRESS による定義 &gt;                  ENTITY track_bleded_solid_with_end_conditions                  SUBTYPE OF (track_bleded_solid);                  end_conditions : LIST [2:2] OF blend_end_condition_select;                  WHERE                  WR1: SIZEOF(TYPEOF(self)                  ['SOLID_SHAPE_ELEMENT_SCHEMA.SOLID_WITH_                  CONSTANT_RADIUS_EDGE_BLEND',                  'SOLID_SHAPE_ELEMENT_SCHEMA.SOLID_WITH_                  VARIABLE_RADIUS_EDGE_BLEND',                  'SOLID_SHAPE_ELEMENT_SCHEMA.SOLID_WITH_                  CHAMFERED_EDGES']) = 1;                  WR2: NOT (('GEOMETRY_SCHEMA.VERTEX' IN TYPEOF(end_conditions[1]))                  AND ('GEOMETRY_SCHEMA.VERTEX' IN                  TYPEOF(end_conditions[2]]));                  WR3: NOT (('GEOMETRY_SCHEMA.VERTEX' IN TYPEOF(end_conditions[1]))                  AND (NOT ((end_conditions[1]                  := SELF¥edge_bleded_solid.blended_edges[1]_edge_start)                  XOR (end_conditions[1]                  := SELF¥edge_bleded_solid.blended_edges[1]_edge_end))));                  WR4: NOT (('GEOMETRY_SCHEMA.VERTEX' IN TYPEOF(end_conditions[2]))                  AND (NOT ((end_conditions[2]                  := SELF¥edge_bleded_solid.blended_edges[HIINDEX(                  SELF¥edge_bleded_solid.blended_edges)].rdge_start)                  XOR (end_conditions[2]                  := SELF¥edge_bleded_solid.blended_edges[HIINDEX(                  SELF¥edge_bleded_solid.blended_edges)].rdge_end))));                  END_ENTITY;</p>	

<条件>

本エンティティは複雑なインスタンスとして利用される。

### 3-5-3 スケッチ

本項では、スケッチの仕様として、Part108 のエンティティを列挙し、その説明、定義および条件を示す。各エンティティ名に記載される括弧内の番号は、ISO 規格書 (ISO10303-108) の該当項を表す。

#### (1) Parameterization スキーマ

##### (A) 変数モデル

エンティティ名	model_parameter (4.4.1)
<説明> 変数モデルに関するエンティティ。	
< EXPRESS による定義 > ENTITY model_parameter ABSTRACT SUPERTYPE OF (ONEOF (bound_model_parameter, unbound_model_parameter)) SUBTYPE OF (variational_representation_item, maths_variable); parameter_description : OPTIONAL text; parameter_current_value : maths_value; DERIVE SELF¥maths_variable.name : label := SELF¥representation_item.name; WHERE WR1 : member_of(parameter_current_value, SELF¥maths_variable.values_space); END_ENTITY;	
<条件> モデルパラメータのインスタンスの値は SELF/math_s variable.values_space に よって指定されたドメインとなる必要がある。	

##### (B) モデルの属性と関連のある変数モデル

エンティティ名	bound_model_parameter (4.4.2)
<説明> モデルの属性と関連のある変数。	
< EXPRESS による定義 > ENTITY bound_model_parameter SUBTYPE OF (model_parameter); DERIVE SELF¥model_parameter.parameter_current_value : maths_value := ?; WHERE WR1 : 'PARAMETERIZATION_SCHEMA.BOUND_PARAMETER_Environment' IN TYPEOF(SELF¥generic_variable.interpretation); END_ENTITY;	

<条件>

bound model parameter のすべてのインスタンスは、 type bound parameter environment のインスタンスを参照する。

(C) モデルの属性と関連のない変数モデル

エンティティ名	unbound_model_parameter (4.4.3)
---------	---------------------------------

<説明>

モデルの属性と関連のない変数。

< EXPRESS による定義 >

```
ENTITY unbound_model_parameter
  SUBTYPE OF (model_parameter);
WHERE
  WR1: 'PARAMETERIZATION_SCHEMA.UNBOUND_PARAMETER_
ENVIRONMENT'
  IN TYPEOF(SELF¥generic_variable.interpretation);
END_ENTITY;
```

<条件>

unbound model parameter のすべてのインスタンスは、 unbound parameter environment のインスタンスによって参照される。

(2) Variational representation スキーマ

(A) 拘束情報やパラメータの関連付け

エンティティ名	invalidate_vrep_item (6.4.1)
---------	------------------------------

<説明>

拘束情報やパラメータを関連付ける。

< EXPRESS による定義 >

```
FUNCTION invalidate_vrep_item(item : variational_representation_item)
: BOOLEAN;
  LOCAL
    reps : SET[1:?] OF representation := using_representations(item);
    svri : SET[1:?] OF variational_representation_item;
    iar : instance_attribute_reference;
    i : INTEGER;
    n : INTEGER := HIINDEX(reps);
  END_LOCAL;
  IF ('PARAMETERIZATION_SCHEMA.BOUND_MODEL_PARAMETER' IN
  TYPEOF(item))
  THEN
    IF 'PARAMETERIZATION_SCHEMA.INSTANCE_ATTRIBUTE_
  REFERENCE'
    IN TYPEOF(item%generic_variable.interpretation.semantics)
    THEN
      BEGIN
        iar := item%generic_variable.interpretation.semantics;
        IF (reps <> using_representations(iar.owning_instance))
        THEN
          RETURN(TRUE);
        END_IF;
      END;
      ELSE RETURN(TRUE); -- parameter not attached to an instance
  attribute
      END_IF;
    END_IF;

    IF ('PARAMETERIZATION_SCHEMA.UNBOUND_MODEL_
  PARAMETER' IN TYPEOF(item))
    THEN
      BEGIN
        REPEAT i := 1 TO n;
          svri := QUERY(q <= reps[i].items |
            'EXPLICIT_CONSTRAINT_SCHEMA.FREE_FORM_
  CONSTRAINT' IN TYPEOF(q));
          IF SIZEOF(QUERY(r <= svri |
  item IN (r.reference_elements + r.constrained_elements))) = 0
          THEN
            RETURN(TRUE);
          END_IF;
        END_REPEAT;
      END;
    END_IF;

    IF
  ('PARAMETERIZATION_SCHEMA.FIXED_INSTANCE_ATTRIBUTE_SET'
  IN TYPEOF(item))
    THEN
      REPEAT i := 1 TO SIZEOF(item.fixed_attributes);
        IF (reps <> using_representations(item.fixed_attributes[i]))
        THEN
          RETURN(TRUE);
        END_IF;
      END_REPEAT;
    END_IF;
  END;
```

```

    END_REPEAT;
  END_IF

  IF ('EXPLICIT_CONSTRAINT_SCHEMA.EXPLICIT_CONSTRAINT' IN
  TYPEOF(item))
  THEN
    IF SIZEOF(QUERY(q <*
    (item.reference_elements + item.constrained_elements) |
    reps <> using_representations(q))) > 0
    THEN
      RETURN(TRUE);
    END_IF;
  END_IF;

  IF ('VARIATIONAL_REPRESENTATION_SCHEMA.
  AUXILIARY_GEOMETRIC_REPRESENTATION_ITEM' IN
  TYPEOF(item))
  THEN
    BEGIN
      REPEAT i := 1 TO n;
        svri := QUERY(q <* reps[i].items |
        'EXPLICIT_GEOMETRIC_CONSTRAINT_SCHEMA.
        EXPLICIT_GEOMETRIC_CONSTRAINT' IN TYPEOF(q));
        IF SIZEOF(QUERY(r <* svri |
        item IN r.reference_elements)) = 0
        THEN
          RETURN(TRUE);
        END_IF;
      END_REPEAT;
    END;
  END_IF;
  RETURN(FALSE); -- no invalid cases have been found
END_FUNCTION;

```

```

IF
('PARAMETERIZATION_SCHEMA.FIXED_INSTANCE_ATTRIBUTE_SET'
 IN TYPEOF(item))
THEN
  REPEAT i := 1 TO SIZEOF(item.fixed_attributes);
    IF (reps <> using_representations(item.fixed_attributes[i]))
    THEN
      RETURN(TRUE);
    END_IF;
  END_REPEAT;
END_IF

IF ('EXPLICIT_CONSTRAINT_SCHEMA.EXPLICIT_CONSTRAINT' IN
TYPEOF(item))
THEN
  IF SIZEOF(QUERY(q < *
(item.reference_elements + item.constrained_elements) |
reps <> using_representations(q))) > 0
THEN
    RETURN(TRUE);
  END_IF;
END_IF;

IF ('VARIATIONAL_REPRESENTATION_SCHEMA.
AUXILIARY_GEOMETRIC_REPRESENTATION_ITEM' IN
TYPEOF(item))
THEN
  BEGIN
    REPEAT i := 1 TO n;
      svri := QUERY(q < * reps[i].items |
'EXPLICIT_GEOMETRIC_CONSTRAINT_SCHEMA.
EXPLICIT_GEOMETRIC_CONSTRAINT' IN TYPEOF(q));
      IF SIZEOF(QUERY(r < * svri |
item IN r.reference_elements)) = 0
      THEN
        RETURN(TRUE);
      END_IF;
    END_REPEAT;
  END;
END_IF;
RETURN(FALSE); -- no invalid cases have been found
END_FUNCTION;

```

<条件>

- **bound model parameter** のインスタンスは、モデルパラメータが参加する全ての表現によって使われたインスタンスの属性に制約される。
- **unbound model parameter** のインスタンスがモデルパラメータが参加する全ての表現で生じるリファレンス要素あるいは制限された要素として少なくとも1つの **free form constraint** のインスタンスに属する。
- **fixed instance attribute set** のインスタンスが固定されているインスタンスの属性セットが参加する全ての表現によって使われないどんなインスタンスの属性も参照しない。
- **explicit constraint** のインスタンスが明示的な制約が属する全ての表現によって使われるインスタンスではない参照された要素あるいは制限された要素を指定しない。
- **auxiliary geometric representation item** のインスタンスが付加的で幾何学的な代表条項が属するすべての代表のために、少なくとも1つの **explicit geometric constraint** で、参照される要素として属する。

### (3) Sketch スキーマ

#### (A) 平面上の点のスケッチ

エンティティ名	implicit_point_on_plane (8.4.1)
---------	---------------------------------

<説明>

平面上の点のスケッチを定義する。

< EXPRESS による定義 >

```
ENTITY implicit_point_on_plane
  ABSTRACT SUPERTYPE OF (ONEOF
    (implicit_planar_intersection_point,
     implicit_planar_projection_point))
  SUBTYPE OF (point, auxiliary_geometric_representation_item);
  using_sketch : positioned_sketch;
  computed_representation
    : cartesian_point;
DERIVE
  plane_of_point :
    plane := get_plane_of_implicit_geometry(using_sketch);
WHERE
  WR1: (plane_of_point $\forall$ elementary_surface.position.location =
    computed_representation) XOR
    (dot_product(plane_of_point $\forall$ elementary_surface.position.p[3],
    get_relative_direction_2points
    (plane_of_point $\forall$ elementary_surface.position.location,
    computed_representation)) = 0);
  WR2: SIZEOF(TYPEOF(computed_representation) *
    ['SKETCH_SCHEMA.IMPLICIT_POINT_ON_PLANE',
    'GEOMETRY_SCHEMA.POINT_REPLICA']) = 0;
END_ENTITY;
```



<条件>

- `computed point` は代表点と同時に存在するか、あるいはそれが平面上にあるかのどちらかである。
- 暗黙のうちに定義されたポイントの計算された表現は `typeimplicit point on plane` あるいは `point replica` であるべきではない。

### (B) 平面上の曲線のスケッチ

エンティティ名 | `implicit_planar_curve` (8.4.4)

<説明>

平面上の曲線のスケッチを定義する。

< EXPRESS による定義 >

```
ENTITY implicit_planar_curve
  ABSTRACT SUPERTYPE OF (ONEOF
    (implicit_intersection_curve,
     implicit_projected_curve,
     implicit_model_intersection_curve,
     implicit_silhouette_curve))
  SUBTYPE OF (curve, auxiliary_geometric_representation_item);
  using_sketch : positioned_sketch;
  computed_representation : curve;
DERIVE
  curve_plane : plane
    := get_plane_of_implicit_geometry(using_sketch);
WHERE
  WR1: SIZEOF(TYPEOF(computed_representation) *
    ['SKETCH_SCHEMA.IMPLICIT_PLANAR_CURVE',
     'GEOMETRY_SCHEMA.CURVE_REPLICA']) = 0;
END_ENTITY;
```

<条件>

計算結果は、`type implicit planar curve` でも `curve replica` でもない。

### (C) 平面上の交曲線のスケッチ

エンティティ名 | `implicit_intersection_curve` (8.4.5)

<説明>

平面上の交曲線のスケッチを定義する。

< EXPRESS による定義 >

```
ENTITY implicit_intersection_curve
  SUBTYPE of (implicit_planar_curve);
  external_surface : surface;
END_ENTITY;
```

### (D) 3次元座標上のスケッチ

エンティティ名	positioned_sketch (8.4.10)
<p>&lt;説明&gt; 3次元座標上のスケッチを定義する。</p>	
<p>&lt;EXPRESSによる定義&gt;  ENTITY positioned_sketch  SUBTYPE OF (geometric_representation_item);  sketch_basis : sketch_basis_select;  auxiliary_elements :  SET[0:?] OF auxiliary_geometric_representation_item;  WHERE  WR1: NOT (('GEOMETRY_SCHEMA.CURVE_BOUNDED_SURFACE' IN  TYPEOF(sketch_basis)) AND NOT ('GEOMETRY_SCHEMA.PLANE' IN  TYPEOF(sketch_basis.basis_surface)));  WR2: NOT (('TOPOLOGY_SCHEMA.FACE_SURFACE' IN  TYPEOF(sketch_basis)) AND  NOT ('GEOMETRY_SCHEMA.PLANE' IN  TYPEOF(sketch_basis.face_geometry)));  WR3: SIZEOF(QUERY(q &lt;* auxiliary_elements   (SIZEOF(TYPEOF(q) *  ['GEOMETRY_SCHEMA.POINT','GEOMETRY_SCHEMA.CURVE']) =  0))) = 0;  WR4: SIZEOF(QUERY(q &lt;* auxiliary_elements    q¥geometric_representation_item.dim &lt;&gt; 3)) = 0;  END_ENTITY;</p>	
<p>&lt;条件&gt;  ・スケッチの基礎が curve bounded surface のセットの場合、基礎をなす表面は平面である。  ・スケッチの基礎が face surface のセットの場合、表面の幾何学的パターンは平らである。  ・auxiliary elements のセットの全てのメンバは曲線あるいはポイントである。  ・auxiliary elements のセットのメンバは3次元である。</p>	

(E) 位置が明示されたスケッチの関係

エンティティ名	implicit_explicit_positioned_sketch_relationship (8.4.12)
<p>&lt;説明&gt; 位置が明示されたスケッチの関係を定義する。</p>	

< EXPRESS による定義 >

```
ENTITY implicit_explicit_positioned_sketch_relationship
  SUBTYPE OF (representation_item_relationship);
  SELF¥representation_item_relationship.relateing_representation_item
    : repositioned_neutral_sketch;
  SELF¥representation_item_relationship.related_representation_item
    : positioned_sketch;
WHERE
  WR1: SIZEOF(QUERY(q <* using_representations(
    SELF¥representation_item_relationship.related_representation_item) |
    item_in_context(
    SELF¥representation_item_relationship.relateing_representation_item,
    q.context_of_items))) > 0;
END_ENTITY;
```

< 条件 >

The implicit and explicit representations of the positioned sketch は少なくとも 1 つの representation context を共有する。

### 3-5-4 アセンブリ

本項では、アセンブリの交換仕様として、Part109のエンティティを列挙し、その説明、定義および条件を示す。各エンティティ名に記載される括弧内の番号は、ISO規格書（ISO10303-109）の該当項を表す。

#### (1) Assembly feature relationship スキーマ

##### (A) 形状関係とその表現の結合

エンティティ名	shape_aspect_relationship_representation_association (4.4.1)
<説明> 形状関係とその表現の結合を定義する。	

< EXPRESS による定義 >

```
ENTITY shape_aspect_relationship_representation_association;
  represented_shape_aspect_relationship:shape_aspect_relationship;
  representing_representation_relationship : representation_relationship;
WHERE
  WR1: ('REPRESENTATIVE_SHAPE_REPRESENTATION' IN
        TYPEOF(representing_representation_relationship¥
        representation_relationship.rep_1)) AND
        ('REPRESENTATIVE_SHAPE_REPRESENTATION' IN
        TYPEOF(representing_representation_relationship¥
        representation_relationship.rep_2));
  WR2: (represented_shape_aspect_relationship.relateing_shape_aspect IN
        using_shape_aspect_of_shape_representation
        (representing_representation_relationship.rep_1))
        AND(represented_shape_aspect_relationship.related_shape_aspect IN
        using_shape_aspect_of_shape_representation
        (representing_representation_relationship.rep_2));
  WR3: ((find_representative_shape_representation_of_product_definition
        (using_product_definition_of_shape_aspect
        (represented_shape_aspect_relationship.relateing_shape_aspect)).
        context_of_items) :=:
        (find_representative_shape_representation_of_shape_aspect
        (represented_shape_aspect_relationship.relateing_shape_aspect).
        context_of_items)) AND
        ((find_representative_shape_representation_of_product_definition
        (using_product_definition_of_shape_aspect
        (represented_shape_aspect_relationship.related_shape_aspect)).
        context_of_items) :=:
        (find_representative_shape_representation_of_shape_aspect
        (represented_shape_aspect_relationship.related_shape_aspect).
        context_of_items));
  WR4: using_product_definition_of_shape_aspect
        (represented_shape_aspect_relationship.relateing_shape_aspect) <=:
        using_product_definition_of_shape_aspect
        (represented_shape_aspect_relationship.related_shape_aspect);
  WR5: find_assembly_root ([using_product_definition_of_shape_aspect
        (represented_shape_aspect_relationship.relateing_shape_aspect)]) :=:
        find_assembly_root ([using_product_definition_of_shape_aspect
        (represented_shape_aspect_relationship.related_shape_aspect)]);
END_ENTITY; -- shape_aspect_relationship_representation_association
```

< 条件 >

- representing\_representation\_relationship は、representative\_shape\_representation のインスタンスと対の関係である。
- 最初の属性によって指定された shape\_aspect\_relationship の relating\_shape\_aspect が 2 番目の属性によって指定された representing\_representation\_relationship の rep\_1 によって示されて representative\_shape\_representation のインスタンスによって表現される。
- shape\_aspect のインスタンスと product\_definition のインスタンスを表している representative\_shape\_representation の 2 つのインスタンスが context\_of\_items のために同じ値を持つ。

(B) 代表形状の表現

エンティティ名	representative shape representation (4.4.2)
<説明> 代表形状の表現を定義する。	
< EXPRESS による定義 > ENTITY representative_shape_representation SUBTYPE OF (shape_representation); WHERE WR1: unique_in_product_definition (SELF) OR unique_in_shape_aspect (SELF); END_ENTITY; -- representative_shape_representation	
<条件> もし存在しているなら、representative_shape_representation は対応する shape_aspect あるいは product_definition のためにユニークである。	

### 3-5-5 属性

本項では、属性の交換仕様として、Part41 のエンティティを列挙し、その説明、定義および条件を示す。各エンティティ名に記載される括弧内の番号は、ISO 規格書 (ISO10303-41) の該当項を表す。

#### (1) External reference スキーマ

##### (A) 付与する属性の ID

エンティティ名	external_source (4.12.4.1)
<説明> 付与する属性の ID を定義する。	
<EXPRESS による定義> ENTITY external_source; source_id : source_item END_ENTITY;	

##### (B) 属性名、属性の内容、参照する情報、参照される情報

エンティティ名	external source relationship (4.12.4.2)
<説明> 属性名、属性の内容、参照する情報、参照される情報を定義する。	
<EXPRESS による定義> ENTITY external source relationship; name : label; description : text; relating_source : external_source; related_source : external_source; END_ENTITY;	