

テクニカルイラストレーション自動生成システム

3F-8

玉柏和男 安部憲広 北橋忠宏  
大阪大学 産業科学研究所

1. はじめに

テクニカルイラストレーション(以下TIと略)は機械の組み立て方法を人間が直感的に理解しやすいように表現した立体製図である。しかしTIを作成するには多くのコストがかかる。その原因の一つに部品レイアウトの構成が挙げられるが3次元空間における複雑な形状を持つ物体のレイアウトの自動決定方法についてはあまり議論がなされていない。そこで本稿では機械部品のレイアウトを制約処理の枠組みにおいて決定する方法を提案する。

2. TIにおける制約

TIにおける制約は、3次元空間において成立するものと2次元に投影した物体間に成立するものを本稿では取り上げる。

2.1 3次元空間における制約

[制約3D-1] 重なり禁止制約

任意の2物体は互いに重なり合わない。

[制約3D-2] 平行移動制約

物体の平行移動のみで(回転移動を行わずに)完成品の状態になる。

[制約3D-3] 等間隔制約

完成品における3次元位置からTIで表現された3次元位置へ同一直線上を移動する部品集合において、分解後に互いに隣接して配置される部品間の移動距離の差の絶対値(以後これを"間隔"と呼ぶ)はすべて等しい。

2.2 2次元空間における制約

[制約2D-1] 穴の隠蔽禁止制約

穴は隣接する部品によって隠蔽されることがない。

3. システム

すべての3次元の制約([3D-1],[3D-2],[3D-3])と2次元の制約([2D-1])を満足するTIを出力する場合を例にあげながらシステム(図1)の各部について述べる。

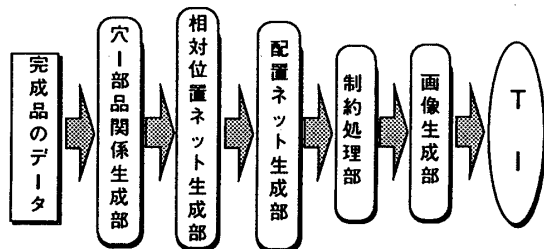


図1 システムの概要

3.1 入力

システムへの入力は完成した状態(組立完了時)における各部品の3次元座標位置(直交座標系)である(図2)。

部品には属性として識別子, 形状, 機能名称, 穴が与えられている。

形状はソリッドモデルで定義する。ソリッドモデルは基本立体(円柱, N角柱, N角錐)の集合演算(和, 差, 積)によって定義される形状モデルである。

本研究では基本立体の和のリスト, 差のリストの繰り返しで形状を定義する<sup>[1]</sup>。

穴は底面と掃引方向が垂直である掃引体であり, 穴の方向はその掃引方向と定義する。掃引方向はx, y, z軸のいずれかに平行とする。穴は両方の底面が開いている穴(両穴)または片方のみが開いている穴(片穴)のいずれかであるとする。また投影方式にはTIにおいて一般的である等角投影図法<sup>[2]</sup>を用いる。

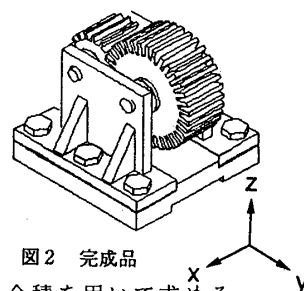


図2 完成品

3.2 穴-部品関係生成部

部品Pのもつ穴Phの中にどの部品が入っているのかをソリッドモデルの集合積を用いて求める。

3.3 相対位置ネット生成部

TIにおいて, ある部品Pの穴Phの中に存在するn個の部品の集合  $P_{in} = \{P_1, \dots, P_n\}$  ( $n \geq 1$ ) はPに対してPhの方向に平行移動されている場合が多い。そこで  $P_{in} \cup \{P\}$  の任意の2要素間に成り立つ穴Phの方向における順序関係を以下のようにして求め, 有向グラフ(相対位置ネット)で表現する。なおグラフの頂点は部品であり, 有向辺  $A \rightarrow B$  はAがBよりもL座標軸に関して+L方向( $L \in \{x, y, z\}$ , 以下同じ)に存在することを表す。

(1)穴を持つすべての部品の集合を  $P_{all}$  とし,  $P_{all}$  が空になるまで  $P_{all}$  から一個の要素Pを取り出し, (2)から(5)まで繰り返す。

(2)  $P$  と  $\forall x \in P_{in}$  とのPhの方向の順序関係を衝突検出を用いて求める。Pと順序関係が成立する  $P_{in}$  の要素を  $P_{in\_ord}$  とする。

(3)もし  $P_{in\_ord} = P_{in}$  ならば(5)へ。  $P_{in\_ord} \neq P_{in}$  ならば(4)へ。

(4)  $\forall x \in P_{in} - P_{in\_ord}$  に対して  $P \rightarrow x$  または  $P \leftarrow x$  のいずれかにする。次に  $P$  と  $\forall y \in U - P_{in}$  との順序関係を衝突検出を用いて求める(ただしUは部品すべての集合, 以下同じ)。

(5)  $P_{in}$  の任意の2要素(部品)間に成立する相対位置関係を衝突検出により求める。

(6)終わり。

衝突検出は部品を平行移動させて部品の非連続面, 断面の交わりを調べる方法<sup>[1]</sup>で行う。衝突検出で部品Aを部品BへL軸の正の向きに平行移動させて衝突する場合,  $B \rightarrow A$  となる。

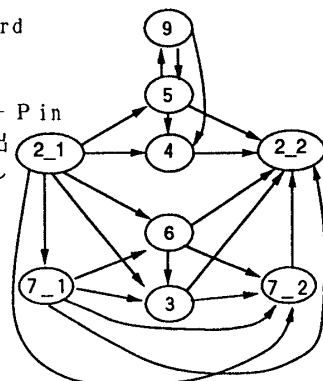


図3 x軸方向相対位置ネット

3.4 配置ネット生成部

ここでは3.3で生成された相対位置ネットおよび相対座標値不変基準に基づき、Uにおいて成立する部品配置に関する半順序関係（配置ネット）を得る。

配置ネットは以下の手順で求める。

- (1)L軸方向相対位置ネットにおいて極大頂点の集合を  $V_{sup}$  とすると  $\forall x \forall y \in V_{sup}$  に対して衝突検出を±L軸方向にそれぞれ行い順序関係があれば有向辺をつける。極小頂点の集合に対しても同様の操作を行う。なお極大（極小）頂点  $v$  とは  $v \rightarrow ve$  ( $v \leftarrow ve$ ) なる  $ve$  が存在しない頂点である。
- (2)L軸方向において  $a \leftarrow b$  かつ  $a \rightarrow b$  なる  $a, b$  が存在する場合、 $a$  と  $b$  はL座標が完成品の状態と相対的に同じであることを意味するので同値類としてまとめる。
- (3)同じ推移的閉包を持つ閉路のない有向グラフのうち辺の数が最小のものに変換する。
- (4)全部品集合Uの要素でグラフの頂点に現れていないものに対して既存の同値類に以下の基準（相対座標値不変基準）で組み入れる。なお、P部品は部品の接続関係を生成する部品で機能名称がネジである部品をさし、N部品はP部品と対になって接続関係を維持する部品であり、被拘束部品はP部品とN部品によって拘束される部品のことである<sup>[1]</sup>。
- （基準1）すべてのP部品と対応するN部品、被拘束部品とはP部品の方向に垂直方向の座標が等しい。
- （基準1'）すべてのP部品とそれに対応する被拘束部品とはP部品の方向に垂直方向の座標が等しい。
- （基準2）ある部品Pの両穴  $Ph$  の内部に存在する部品が  $P_1, \dots, P_n$  ( $n \geq 1$ ) の場合、Pは少なくともある  $P_i$  ( $1 \leq i \leq n$ ) と  $Ph$  の方向と垂直な座標値が不変である同値関係を持つ。

3.5 制約処理部

3.4で得られた配置ネットの有向辺に変数を割当て、それらの変数間に成り立つ連立方程式を生成する（図4、y, z軸方向は略）。次に穴を持つ部品が他の部品によって隠蔽されない最低距離を対応する変数に代入し制約ロジックプログラミング言語CAL<sup>[3]</sup>を利用し解く。値が不定の変数については、その変数に関連のある部品群をユーザに提示することによりユーザが具体値を決定する。

L軸方向の配置ネットへの変数の割当ては以下の方法で行う（用語の定義は表1を参照）。

- (1)すべての極大（小）頂点の集合を  $V_{sup}$  ( $V_{inf}$ ) とすると  $\forall v \in V_{sup}$  ( $V_{inf}$ ) に対してダミーの頂点  $L+$  ( $L-$ ) との間には有向辺  $L+ \rightarrow v$  ( $v \rightarrow L-$ ) を付加する（この処理後の配置ネットをGとする）。
- (2)もしGにおいて  $path(a, b)$  がすべて  $singlePath(a, b)$  である  $paraPaths(a, b)$  が存在すれば(3)へ、存在しなければ(6)へ。
- (3)  $path(a, b)$  ( $\in paraPaths(a, b)$ ) に現れる辺すべてに同一名の変数を割り当てる。
- (4)各辺に割り当てられた変数と  $a$  と  $b$  の間隔を表す変数  $X_{ab}$  との関係を表す連立方程式を生成する。
- (5)  $paraPaths(a, b)$  に現れる辺集合を  $V_p$ 、頂点集合を  $E_p$  とすると、新たに  $G = ((V - V_p) \cup \{a, b\}, (E - E_p) \cup \{a \rightarrow b\})$  とし、辺  $a \rightarrow b$  に変数  $X_{ab}$  を割り当てた後、(2)へ。
- (6)もしGに  $paraPaths(a, b)$  が存在すれば、 $path(a, b)$  に現れる辺に変数を割り当て、連立方程式をたてる。
- (7)変数が割り当てられていない辺に変数を割り当てる。
- (8)終わり。

3.6 画像生成部

制約処理部で得られた各部品の座標値を入力として図を生成する（図5）。

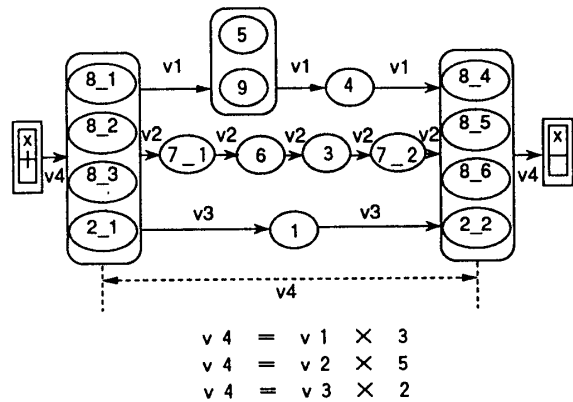


図4 x軸方向配置ネットと変数割り当て

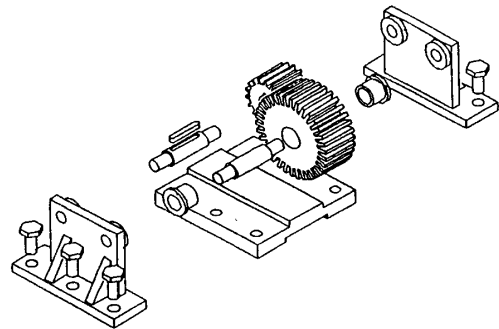


図5 実行結果

☆ $path(a, b)$	: 有向グラフ $G = (V, E)$ に対して頂点の列 $[a, v_1, \dots, v_n, b]$ ( $a, v_1, \dots, v_n, b \in V$ ) において $a \rightarrow v_1, \dots, v_n \rightarrow b$ がすべて $E$ の要素である場合 $[a, v_1, \dots, v_n, b]$ を $path(a, b)$ と定義する。
☆ $paraPaths(a, b)$	: 要素数 ( $ path(a, b) $ ) $\geq 2$ が成立する場合、 $\{path(a, b)\}$ を $paraPaths(a, b)$ と定義する。
☆ 分岐頂点	: $G = (V, E)$ において $v, a, b \in V, a \neq b, v \rightarrow a \in E, v \rightarrow b \in E$ が成り立つ頂点 $v$ 。
☆ 合流頂点	: $G = (V, E)$ において $v, a, b \in V, a \neq b, a \rightarrow v \in E, b \rightarrow v \in E$ が成り立つ頂点 $v$ 。
☆ $singlePath(a, b)$	: $path(a, b) = [a, v_1, \dots, v_n, b]$ に現れる頂点の集合を $V_p$ とし、 $\forall v \in V_p - \{a, b\}$ が分岐頂点、合流頂点のいずれでもない $path(a, b)$ 。

表1 用語の定義

4. まとめ

TIにおいて3次元、2次元における制約の一部を満足する部品配置を得ることができた。

今後は部品のクラス・機能に依存した2次元における制約について検討していく予定である。

謝辞

本研究において制約処理系CAL、EUS-LISPを快く使用させて頂きましたICOTの相場亮氏、電総研の松井俊浩氏に深く感謝いたします。

参考文献

- [1] 山田誠二, 安部憲広, 辻三郎: 電気ドリル分解・組立てコンサルタント・システム, 人工知能学会誌, Vol1 No.1, pp.116-123(1986).
- [2] 近藤謙: 作図と解説テクニカルイラストレーション, 三共出版(1976)
- [3] 相場亮: 「これからの高レベル言語は?」, 数理科学, No.313, pp.50-57(1989)