

# ロボットによる実行を目的とした人の組み立て作業の理解 接触状態遷移からの最適軌道の生成

高松 淳<sup>\*1</sup> 小川原 光 一<sup>\*1</sup> 木村 浩<sup>\*2</sup> 池内 克史<sup>\*1</sup>

## Understanding of Human Assembly Tasks for Robot Execution Generation of Optimal Trajectories Based on Transitions of Contact Relations

Jun Takamatsu<sup>\*1</sup>, Koichi Ogawara<sup>\*1</sup>, Hiroshi Kimura<sup>\*2</sup> and Katsushi Ikeuchi<sup>\*1</sup>

The *planning-from-observation* paradigm is widely noticed as a novel robot-programming technique. It consists of two parts: (1) recognition of a human demonstration from observation as symbolic representation, i.e., a sequence of movement primitives and (2) execution of the same task. Symbolic representation enables a robot to achieve the same task even in a different environment. We already proposed a method to build a symbolic representation for an assembly task. However, for a robot to execute the task, it is necessary to adjust parameters of each movement primitive to generate appropriate trajectories of objects, especially a path which maintains one of any contact states. Many researchers have proposed methods to calculate such a path using a non-linear optimization method, for example, the potential field method, the probabilistic method, and so on. Although these methods are very powerful as calculation tools on a computer, their solutions are not optimal. We propose a novel method to calculate such an optimal path using a linear solution.

**Key Words:** Planning from Observation, Assembly Tasks, Manipulation Skills, Path Planning

### 1. はじめに

ロボットの動作プログラムを簡便に生成する方法として、いわゆる「見まねによる行動獲得」[1]~[10]の手法が近年注目されている。その手法の大きな特徴は、ロボットに行わせたい作業を視覚等のセンサを用いて獲得し、抽象表現に変換して記述することである。実行時に環境に合わせて目的の動作を実現するプログラムを自動生成するには、この抽象化された表現が必要不可欠である。これにより、ロボットは教示のときとは異なる環境下でも目的の動作を実現できる柔軟性を獲得することができる。我々もこのコンセプトに基づき、接触状態遷移モデルから組み立て作業に必要な不可欠な動作要素である「サブスキル」を導出し、これを用いて目的とする組み立て作業を表現する方法を提案している[11]。

しかし「見まねによる行動獲得」の手法において、獲得した作業を再現する際、実行時の環境に合わせて抽象的な表現を対象物の軌道などの具体的な動作に変換する必要が生じる。本論文では、この軌道生成問題について論じる。

組み立て作業の軌道生成においては、ある特定の接触状態を維持したまま別の接触状態に遷移する軌道を生成することが要求される。McCarragher と Asada は、平面運動において与えられた接触状態間の遷移を満たす軌道を求める方法を提案し、さらに解となる軌道が複数個ある場合、より成功率の高い解を選び出す方法を提案している[12]。Hirukawa は、三次元多面体同士の接触状態間の遷移を実現する軌道が満たす条件の定式化の方法、および解析的な解法を示している[13]が、6変数の非線形方程式を解く必要があり、非常に困難である。特に、定式化の仕方により解の導出の困難さが変化することも示されており、効果的に適用するためには知識と経験を要する。

その困難さを解決するために、Ji と Xiao は適切な座標系を設定し、Probabilistic Roadmap Method (PRM) [14]と呼ばれる乱数を用いる手法により軌道を計算する方法を提案している[15]。ただし解となる軌道が複数個ある場合、ある目的に最も合致した最適軌道を求めることは非常に困難である。

本論文では、軌道生成の際に回転軸の向きを一定にするという拘束を加えて、非線形方程式を解くことなく線形解法のみで最適な解のみを求める方法を新たに提案する。この拘束は人間が組み立て作業で行う行動を妨げるものではない。その際、以下のような仮定を設ける：

- (1) 組み立て作業で扱うすべての多面体物体の形状は既知であると、変形しない。

原稿受付 2003年5月21日

<sup>\*1</sup>東京大学

<sup>\*2</sup>電気通信大学

<sup>\*1</sup>The University of Tokyo

<sup>\*2</sup>The University of Electro-communications

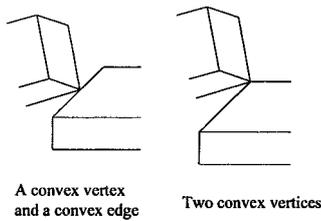


Fig. 1 Two types of singular contact-elements

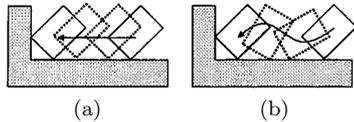


Fig. 2 (a) Optimal trajectory and (b) not optimal trajectory

- (2) 把持された物体（以後、把持物体と記述する）のみが移動し、それ以外の物体（以後、環境物体と記述する）は静止している。
- (3) 接触状態遷移を発生させる軌道は、並進運動、および回転軸の向きが変わらない回転運動のみで実現され、それ以外については扱わない。
- (4) 特異な接触（Fig. 1 参照）を維持したまま把持物体を動かすことはない<sup>†</sup>。
- (5) 求められた最適軌道上には障害物は存在しない。

ここで最適解とは、拘束されていない自由度のパラメータに関して変化のない軌道のことを指す。具体的に Fig. 2 に示す二つの接触状態を遷移する場合、紙面垂直方向の回転は遷移前後でまったく拘束されないため、定義より最適軌道は Fig. 2 (a) のようになり、(b) は最適ではない。PRM のような確率的手法では必ずしも (a) のような解が得られるとは限らない。最適性については、3 章の最適軌道生成法導出の際に改めて定式化する。

本論文は以下のように構成されている。2 章では、最適軌道生成のために必要となる微小変位解析について述べる。3 章では、微小変位解析の結果を用いて把持物体の最適軌道を生成する方法を述べる。4 章では、本手法に対する問題点とその解決法を示す。5 章ではロボットによる実装の試みについて、6 章では過去の研究との比較、7 章では本論文のまとめを、8 章で今後の課題について述べる。

## 2. 微小変位解析

我々は、「スキルに基づくマニピュレーションシステム」[16] の考え方に基づき、組み立て作業中に生じる接触状態遷移から、組み立て作業に必要な不可欠な動作プリミティブ（これをサブスキルと呼ぶ）の連続で組み立て作業を記述する方法を提案している [11]。具体的には、1. 接触状態を物体のとりうる微小変位に基づいて分類し、2. その遷移に伴う微小変位の変化から、対応するサブスキルを選択する。という 2 ステップからなる。こ

こでは 1. の微小変位解析について、特に最適軌道を導出するために必要となる式の導出について述べる。

比留川らは多面体物体間の接触における物体の可能な微小変位を幾何モデルから導出する方法を提案している [17]。それによると、可能な微小変位は式 (1) の形で表現することができる。ただし、 $\mathbf{F}_{ij} \in R^3$  は分離面の法線（例えば、把持物体の点と環境物体の面が接触している場合、 $\mathbf{F}_{ij}$  は面の外向き法線に等しい）を表し、 $\Delta \mathbf{X} \in R^3, \Omega \in R^3$  はそれぞれ位置および姿勢の微小変位を、 $N$  は接触点の数、 $M(i)$  は  $i$  番目の接触点における分離面の数を表す。また  $J_i \in R^{3 \times 6}$  は  $i$  番目の接触点の微小変位と物体の位置姿勢の微小変位との関係を表すヤコビ行列である。

$$\bigcap_i^N \bigcup_j^{M(i)} \mathbf{F}_{ij}^T J_i \begin{pmatrix} \Delta \mathbf{X} \\ \Omega \end{pmatrix} \geq 0 \quad (1)$$

本論文では、微小変位を screw 表現 [18] を用いて表す。screw 表現では、微小変位をある軸に沿った並進運動とそれと同じ軸を中心とする回転運動の組み合わせとして表現する。具体的には、世界座標系で軸の向きを  $\mathbf{S}_0$ 、軸の位置を  $\mathbf{P}$ 、回転運動に対する並進運動の割合を  $p$  とすると、その微小変位は、六次元ベクトル  $[\mathbf{S}_0, \mathbf{S}_1]$  として表現される。ただし、screw 成分  $\mathbf{S}_1$  は  $\mathbf{S}_1 = \mathbf{P} \times \mathbf{S}_0 + p\mathbf{S}_0$  を満たす。 $p = 0$  のときは純粋な回転運動を表し、 $p = \infty$  のとき、つまり screw 成分が  $\lim_{p \rightarrow \infty} \begin{bmatrix} \mathbf{S}_0 \\ p, \mathbf{S}_1 \end{bmatrix} = [\mathbf{0}, \mathbf{S}_0]$  の時は、純粋な並進運動を表す。

実際に screw 表現を用いると、式 (1) は式 (2) のように書き換えられる。ただし、 $\mathbf{P}_i$  は  $i$  番目の接触点の位置を表す [18]。

$$\bigcap_i^N \bigcup_j^{M(i)} \mathbf{F}_{ij} \cdot \mathbf{S}_1 + (\mathbf{P}_i \times \mathbf{F}_{ij}) \cdot \mathbf{S}_0 \geq 0 \quad (2)$$

最適軌道生成の際には、ある接触状態を維持し続ける軌道を生成することが要求される。その場合、複数分離面が存在する場合でも、各分離面上を接触点が動く必要があるため、可能な微小変位は式 (3) で表すことができる。

$$\bigcap_i^N \bigcap_j^{M(i)} \mathbf{F}_{ij} \cdot \mathbf{S}_1 + (\mathbf{P}_i \times \mathbf{F}_{ij}) \cdot \mathbf{S}_0 = 0 \quad (3)$$

$\cup$  が  $\cap$  に変わることにより、式 (3) は単なる連立線形方程式となる。式 (3) の Rank を全束縛自由度と定義する。

Screw 表現では、並進微小変位は  $\mathbf{S}_0 = \mathbf{0}$  で表されるため、式 (3) に  $\mathbf{S}_0 = \mathbf{0}$  を代入して得られる式 (4)<sup>††</sup> は可能な並進微小変位を表す。

$$\bigcap_i^n \mathbf{F}_i \cdot \mathbf{S}_1 = 0 \quad (4)$$

式 (4) の Rank を並進束縛自由度と定義する。

同様に回転軸の向きは  $\mathbf{S}_0$  で表されるため、線形独立でない  $\mathbf{F}_{ij}$  の全組み合わせを発見し  $\mathbf{S}_1$  を消去することにより、 $\mathbf{S}_0$  のみからなる、とりうる回転軸の向きを表す方程式を得ることができる。具体的には線形独立でない  $\mathbf{F}_{ij}$  の組み合わせを改めて  $\{\mathbf{H}_1, \dots, \mathbf{H}_m\}$  とすると、式 (5) が成り立つ。ただし  $\forall i, c_i \neq 0$  である。

<sup>†</sup> 本論文の目的は、人間の組み立て作業を理解することである。我々も組み立て動作の際このような動作を避けることから、この仮定は妥当なものである。

<sup>††</sup> 記述を簡単にするために、二つの  $\cap$  を一つにまとめてある。

$$\sum_i^m c_i \mathbf{H}_i = \mathbf{0} \quad (5)$$

式 (5) の両辺に  $\mathbf{S}_1$  との内積をとることにより, 式 (6) が得られる.

$$\sum_i^m c_i \mathbf{H}_i \cdot \mathbf{S}_1 = \mathbf{0} \quad (6)$$

式 (6) に式 (3) を代入し, さらに全組み合わせに対して同様な手順を繰り返すことにより, とりうる回転軸の向きを表す式 (7) を得ることができる.

$$\bigcap_i^n \mathbf{G}_i \cdot \mathbf{S}_0 = \mathbf{0} \quad (7)$$

式 (7) の Rank を回転束縛自由度と定義する.

### 3. 最適軌道生成法

本章では, 二つの直接遷移可能な接触状態, およびその接触状態を満たす物体の位置姿勢が与えられたとき, その接触状態遷移を実現する把持物体の軌道を生成する方法について述べる. ここでは, 遷移前に比べて遷移後の接触状態の全束縛自由度の方が大きいとする. もしそうでない場合には, 逆の遷移を実現する軌道を生成し, その軌道を逆にすることにより目標軌道を生成することができる. 簡単のため, 世界座標系と環境物体座標系は一致しているものとし, 特に記述のない場合は世界座標系で表されているものとする.

与えられた二つの接触状態は直接遷移可能であるので, 接触状態遷移前の並進, 回転束縛自由度は, 遷移後のそれらの自由度以下でなくてはならない. 回転束縛自由度が遷移前後で変わらないとき, 並進変位のみで表される軌道が最適軌道であると定義する. もし回転束縛自由度が遷移後に増加する場合は, 最適軌道に回転変位を含む. 以下では前者の並進変位の場合と, 後者の回転変位の場合に分けて最適軌道生成法を導出する.

#### 3.1 並進変位

##### 3.1.1 最適軌道の定義

ある接触状態  $C_s$  からある接触状態  $C_e$  に並進変位のみで遷移する場合について考える. 今, 両接触状態を満たす把持物体の位置姿勢  $\mathbf{q}_s = (\mathbf{t}_s, \Theta)$ ,  $\mathbf{q}_e = (\mathbf{t}_e, \Theta)$  が与えられたとする. ただし,  $\mathbf{t}_* (\in R^3)$  は位置を,  $\Theta (\in SO(3): 3 \times 3 \text{ 直交行列})$  は姿勢を表す.

このとき, それぞれの接触状態における可能な変位  $\Delta \mathbf{t} (\in R^3)$  は前述のとおり式 (8), (9) のような線形連立方程式で表すことができる. ただし  $F_s \in R^{l \times 3}$ ,  $F_e \in R^{m \times 3}$  であるとする.

$$F_s \Delta \mathbf{t} = \mathbf{0} \quad (8)$$

$$F_e \Delta \mathbf{t} = \mathbf{0} \quad (9)$$

並進変位のみでよい場合, 物体が多面体であるという条件より, 直線軌道で現在の接触状態を維持しながら次の接触状態へ遷移することができるので, 物体の軌道は関数  $\mathbf{q}(s) = (\mathbf{t}(s), \Theta(s)) (0 \leq s \leq 1)$  で与えられる. ただし,  $\mathbf{t}(s) = s \Delta \mathbf{t}_d + \mathbf{t}_s$ ,  $\Theta(s) = \Theta$  であり, かつ  $\mathbf{q}(s) (0 \leq s < 1)$  は接触状態  $C_s$  を  $\mathbf{q}(1)$  は接触状態  $C_e$  を満たす物体の位置姿勢であるとする.

Table 1 Classification based on rank

Rank $F_s$	Rank $F_e$	
0	3	} Case 1.
1		
2		
0	2	} Case 2.
1		
0	1	} Case 3.

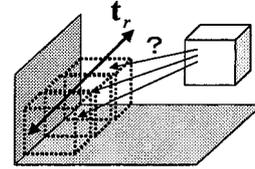


Fig. 3 Case 2: Redundant solution in translation

このとき, 式 (10) を満たす軌道を最適軌道であるとする. ただし,  $\mathbf{T}_e$  は式 (9) の解領域を表す.

$$\mathbf{1} \cdot \Delta \mathbf{t}_d = 0 \quad (\forall \mathbf{1} \in \mathbf{T}_e) \quad (10)$$

この最適性は並進変位量を小さくする傾向がある.

##### 3.1.2 最適軌道生成

最適軌道を生成するためには,  $\Delta \mathbf{t}_d$  を決定してやればよい. 接触状態  $C_s$  を満たしながら変位するためには,  $\Delta \mathbf{t}_d$  は式 (8) を満たす必要がある. 束縛自由度の条件より,  $\text{Rank } F_s < \text{Rank } F_e$  が成り立つので, それぞれの Rank のとり得る値の組み合わせは Table 1 に示す 6 種類が存在する. ここではそれらを三つの場合に分けて, 最適軌道生成を定式化する.

**Case 1.** Rank  $F_s = \{0, 1, 2\}$ , Rank  $F_e = 3$

Rank  $F_e = 3$  より, 遷移後の接触状態を満たす把持物体の位置は一意に決定される. これより  $\Delta \mathbf{t}_d = \mathbf{t}_e - \mathbf{t}_s$  となる.

**Case 2.** Rank  $F_s = \{0, 1\}$ , Rank  $F_e = 2$

Fig. 3 に,  $F_s = 0, F_e = 2$  のときの具体的な例を示すが, Rank  $F_e = 2$  より  $\mathbf{T}_e$  は一次元なので,  $\exists \mathbf{t}_r \in \mathbf{T}_e - \{\mathbf{0}\}$  に対して, 式 (11) を満たせば最適軌道の条件である式 (10) を満たす.

$$\Delta \mathbf{t}_d \cdot \mathbf{t}_r = 0 \quad (11)$$

遷移後の把持物体の位置姿勢は  $\mathbf{q} = (\mathbf{t}_e + u \mathbf{t}_r, \Theta) (u \in R)$  と表すことができ, 求めるべき変位は式 (12) で表される.

$$\Delta \mathbf{t}_d = \mathbf{t}_e + u \mathbf{t}_r - \mathbf{t}_s \quad (12)$$

式 (12) を式 (11) に代入することにより,

$$u = \frac{(\mathbf{t}_s - \mathbf{t}_e) \cdot \mathbf{t}_r}{|\mathbf{t}_r|^2}$$

が得られ, これより  $\Delta \mathbf{t}_d$  を得ることができる.

**Case 3.** Rank  $F_s = 0$ , Rank  $F_e = 1$

今, 式 (9) の解の基底を  $\{\Delta \mathbf{t}_1, \Delta \mathbf{t}_2\}$  とする. このとき, 最適軌道の条件式 (10) より,  $\Delta \mathbf{t}_d$  は式 (13) と表すことができる (Fig. 4 参照).

$$\Delta \mathbf{t}_d = u(\Delta \mathbf{t}_1 \times \Delta \mathbf{t}_2) \quad (13)$$

遷移後は接触状態  $C_e$  を満たすことを用いて,  $u$  を決定することができ, その結果  $\Delta \mathbf{t}_d$  を決定することができる.

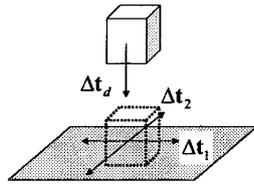


Fig. 4 Case 3: Redundant solution in translation

3.2 回転変位

次に、回転変位における最適軌道生成法について述べる。通常、回転変位における軌道生成は、非線形冗長連立方程式を解く必要があり、困難な問題である。しかし、回転軸の向きが変わらない運動のみを考えたとき、微小変位解析より回転軸の向きに関する関係式が求められているので、先に回転軸の向きと回転量を求め、次に求めた回転軸回りの回転量と物体の位置に関する関係式を解くことにより、この問題を解決することができる。

3.2.1 最適軌道の定義

ある接触状態  $C_s$  からある接触状態  $C_e$  に回転変位で遷移する場合について考える。今、接触状態を満たす物体の位置姿勢をそれぞれ  $\mathbf{q}_s = (\mathbf{t}_s, \Theta_s)$ ,  $\mathbf{q}_e = (\mathbf{t}_e, \Theta_e)$  とする。

このとき、それぞれの接触状態における可能な回転軸の向き  $\mathbf{a} (\in R^3)$  は、並進変位の時と同様に、式 (14), (15) のような連立方程式の形で表すことができる。

$$G_s \mathbf{a} = \mathbf{0} \tag{14}$$

$$G_e \mathbf{a} = \mathbf{0} \tag{15}$$

物体の軌道が関数  $\mathbf{q}(s) = (\mathbf{t}(s), \Theta(s)) (0 \leq s \leq 1)$  で与えられたとする。  $\Theta(s)$  は軸  $\mathbf{a}_d (\in R^3)$  を中心とした等速回転運動、つまり式 (16) で表されているものとする。ただし、  $\mathbf{t}(s) \in R^3$ ,  $R(\mathbf{a}, \theta) (\in SO(3))$  は、軸  $\mathbf{a}$  の回りに  $\theta (\in R)$  回転する座標変換を表す。

$$\Theta(s) = R(\mathbf{a}_d, s\theta_d)\Theta_s \tag{16}$$

また、  $\mathbf{q}(s) (0 \leq s < 1)$  は接触状態  $C_s$  を  $\mathbf{q}(1)$  は接触状態  $C_e$  を満たす物体の位置姿勢であるとする。

このとき、式 (17) を満たす軌道を最適軌道であるとする。ただし、  $\mathbf{A}_e$  は式 (15) の解領域を表す。

$$\mathbf{l} \cdot \mathbf{a}_d = 0 \quad (\forall \mathbf{l} \in \mathbf{A}_e) \tag{17}$$

並進の場合と同様に、この最適性は回転変位量を小さくする傾向がある。

3.2.2 回転軸の向き  $\mathbf{a}_d$  と回転量  $\theta_d$  の決定

並進の場合と同様に、行列  $G_s, G_e$  の Rank の値に基づく 6 種類の組み合わせを四つの場合に分けて、最適軌道生成を定式化する。

Case 1. Rank  $G_s = \{0, 1, 2\}$ , Rank  $G_e = 3$

並進の場合と同様に、遷移後の接触状態を満たす保持物体の姿勢は一意に決定される。これより求めるべき回転軸の向き  $\mathbf{a}_d$  と回転量  $\theta_d$  は、  $\Theta_s$  と  $\Theta_e$  より求めることができる。

Case 2. Rank  $G_s = 0$ , Rank  $G_e = 2$

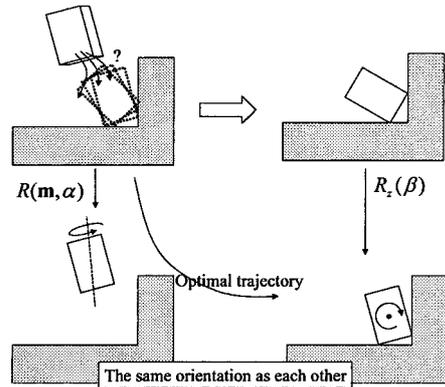


Fig. 5 Case 2: Redundant displacement in rotation

Rank  $G_e \neq 3$  より、Fig. 5 に示すように、遷移後の接触状態を満たす姿勢は一意に決定できない。冗長な部分を取り除くことにより最適な回転軸を求める。

式 (15) の解の正規基底を  $\{\mathbf{a}_1\}$  とし、式 (14) の解の正規直交基底を、一部重なるように  $\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}$  とする。基底  $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$  が、それぞれ  $y, z, x$  軸となるような直交座標系  $\Sigma_a$  を考える。その座標系において、遷移前後の物体の姿勢が  ${}^a\Theta_s, {}^a\Theta_e \in SO(3)$  と表されたとする。

条件より  $\mathbf{a}_1$  軸向きの回転変位は、接触状態  $C_e$  を維持することができるので、Fig. 5 に示すように、以下に示すステップを経て、姿勢  ${}^a\Theta_e$  から  ${}^a\Theta_s$  へ変位することができる：

- (1)  $z$  軸を中心に接触状態  $C_e$  を維持しながら回転
  - (2)  $xy$  平面上の適当な軸  $\mathbf{m}$  を中心に回転
- このことより、式 (18) が成り立つ。ただし、  $R_*(\theta) (\in SO(3))$  は  $*$  軸回りに  $\theta$  回転した座標変換を表す。

$$R(\mathbf{m}, \alpha) {}^a\Theta_s = R_z(\beta) {}^a\Theta_e \tag{18}$$

この式より、  $\alpha, \beta, \mathbf{m}$  を求めることができる。1 番目のステップ、つまり  $R_z(\beta)$  は接触状態遷移には冗長な回転であるので、最適回転変位は  $R(\mathbf{m}, \alpha)$  で表される。

Case 3. Rank  $G_s = 1$ , Rank  $G_e = 2$

式 (15) の解の正規基底を  $\{\mathbf{a}_1\}$  としたとき、Case 2. の方法と同様に、式 (14) の解の正規直交基底を、一部重なるように  $\{\mathbf{a}_1, \mathbf{a}_2\}$  と必ずしもとれるとは限らない。つまり、式 (15) の解領域は、式 (14) の解領域の線形部分空間であるとは限らない。しかし、この節では線形部分空間である場合についてのみ論じ、それ以外の場合の対処法については 4 章で具体的に述べる。基底  $\mathbf{a}_1, \mathbf{a}_2$  がそれぞれ  $y, z$  軸となるような直交座標系  $\Sigma_a$  を考える。その座標系において、遷移前後の物体の姿勢が  ${}^a\Theta_s, {}^a\Theta_e \in SO(3)$  と表されたとする。Case 2. と同様に、Fig. 6 に示すステップを経て、姿勢  ${}^a\Theta_s$  から  ${}^a\Theta_e$  に移ることができることから、式 (19) が成り立つ。

$$R_z(\alpha) {}^a\Theta_s = R_y(\beta) {}^a\Theta_e \tag{19}$$

この式より、  $\alpha, \beta$  を求めることができる。  $R_y(\beta)$  は接触状態遷移には冗長な回転であるので、最適姿勢変位は  $R_z(\alpha)$  で表される。

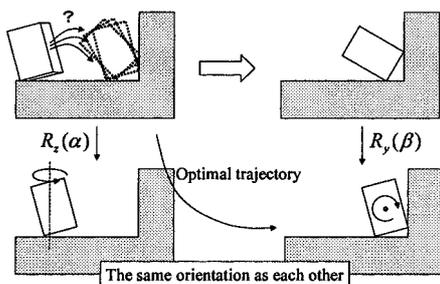


Fig. 6 Case 3: Redundant displacement in rotation

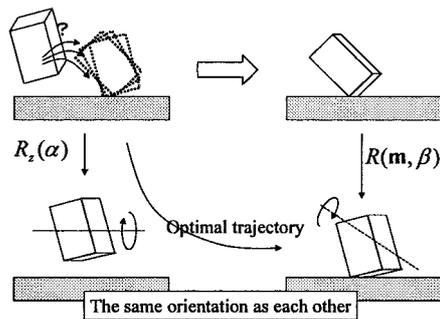


Fig. 7 Case 4: Redundant displacement in rotation

#### Case 4. Rank $G_s=0$ , Rank $G_e = 1$

Case 2. と同様に、式 (15) の解の正規直交基底を  $\{\mathbf{a}_1, \mathbf{a}_2\}$  とし、式 (14) の解の正規直交基底を、一部重なるように  $\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}$  とする。基底  $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$  が、それぞれを x, y, z 軸となるような直交座標系  $\Sigma_a$  を考える。その座標系において、遷移前後の物体の姿勢が  ${}^a\Theta_s, {}^a\Theta_e \in SO(3)$  と表されたとする。先ほどと同様に、Fig. 7 に示すステップを経て、姿勢  ${}^a\Theta_s$  から  ${}^a\Theta_e$  に移ることができることから、式 (20) が成り立つ。ただし、 $\mathbf{m}$  は xy 平面上にあるとする。

$$R_z(\alpha) {}^a\Theta_s = R(\mathbf{m}, \beta) {}^a\Theta_e \quad (20)$$

この式より、 $\alpha, \beta, \mathbf{m}$  を求めることができる。 $R(\mathbf{m}, \beta)$  は接触状態遷移には冗長な回転であるので、最適姿勢変位は  $R_z(\alpha)$  で表される。

#### 3.2.3 回転量と位置姿勢の関係式の導出

把持物体の軌道は、 $\mathbf{q}(s) = (\mathbf{t}(s), \Theta(s))$  で表されているとする。前述の方法により  $\Theta(s)$  は決定されているものとする。ある接触状態を満たしているときの把持物体のとりうる位置姿勢は、非線形連立方程式の形で表すことができる [13]。

$$\bigcap_{i=1}^l f_i(\mathbf{t}(s), \Theta(s)) = 0 \quad (21)$$

式 (21) の  $f_i$  の非線形性は  $\Theta(s)$  のみにより引き起こされていることが分かっている。今ある  $s$  が決定されたとき、 $\Theta(s)$  は定数となるので、 $f_i$  は線形の式となり式 (21) も線形連立方程式となる。

通常、ロボットは離散時間ステップで制御が行われるので、ある  $s$  における  $\mathbf{t}(s)$  の値を計算できれば十分であり、その解を解くことは冗長性はあるものの容易である。

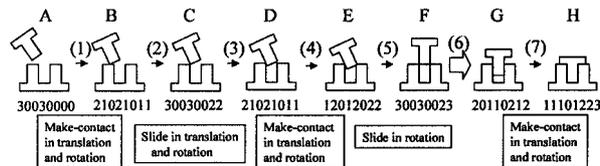


Fig. 8 Result of assembly-task recognition

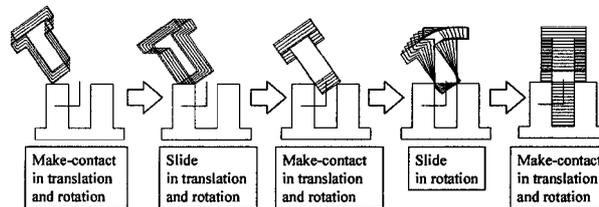


Fig. 9 Optimal trajectory

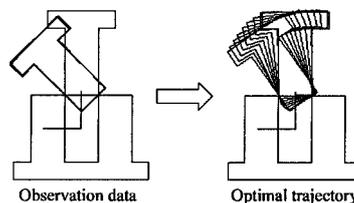


Fig. 10 A trajectory from observation and calculation of an optimal one

### 3.3 最適軌道生成実験 1

提案した計算手法を実際にも実装した。まず文献 [11] で提案された手法を用いて、人が行った Fig. 8 に示す二次元的なペグインサーション作業の接触状態遷移と、各接触状態を満たす物体の位置姿勢を、視覚により得た。各接触状態において運動自由度を計算し、その遷移を調べることで、Fig. 8 に示すサブスキル表現を得た。図の接触状態の下にある数字は、左から並進維持、離脱、拘束、回転維持、離脱、拘束、並進束縛、回転束縛自由度を表す。解析はすべて三次元空間で行われている。

スタートの状態は、回転最適軌道生成の Case 3. で指摘した片方の解がもう片方の線形部分空間になるようにするため、ペグとホルの奥行き方向を合わせた状態とした。各接触状態遷移を引き起こす最適な軌道を計算し、仮想空間で把持物体を動かしてみた結果を、Fig. 9 に示す。

特に Fig. 8 (5) の遷移に対応する部分の視覚による認識結果と軌道生成結果を比較したものを Fig. 10 に示す。視覚による誤差修正では、誤差修正の困難さのため、その軌道上で十分な数の位置姿勢のサンプル点を得ることができなかったが、最適軌道生成によって十分な数のサンプル点を得ることができる。

## 4. 本手法の問題点とその解決法

### 4.1 Screw 理論の限界とその解決法

種々の実験を通じて、回転束縛自由度が 0 から 1, 1 から 3 へ移る際、最適軌道生成に失敗するという問題が発生した。Fig. 11 に回転束縛自由度が 0 から 1 へ移る場合に失敗した例を示す。本節では失敗の原因と解決法を述べる。

回転軸の向きに関する方程式を得る際、非線形の式で表され

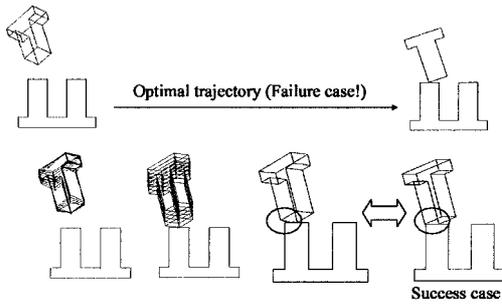


Fig. 11 Fail to calculate an optimal trajectory

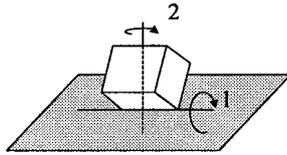


Fig. 12 Edge-face contact

る物体の可能な変位を線形近似して扱っている．一般的に，線形近似により生じる誤差により誤った解が導き出される可能性が示唆されている [19]．本手法においても，遷移前後いずれかにおいて回転束縛自由度が 1 である場合に問題が発生した．

例えば，Fig. 12 に示す辺一面接触において，回転束縛自由度の解析により導き出された条件式は，辺と面の法線を含む平面に回転軸がある際，その回転により接触状態は維持されると解釈してしまう．これは明らかに間違った解釈であり，その結果最適軌道の生成に失敗してしまう．

間違った解釈を解決する方法として，我々は線形ではなく，二次式で可能な変位を近似する方法を提案しており [20]，それを用いることにより，辺と法線方向のみの回転軸 (Fig. 12 の軸 1 と 2) しか接触を維持することができないと正しく解釈することができる．

正しい解釈を用いて最適軌道生成を行う方法を導くことは今後の課題となるが，今回は ad-hoc ではあるが以下のような方法で回転束縛自由度が 1 の際の問題を解決した．

まず，回転束縛自由度が 1 になるのは，たいていの場合辺一面接触か面一面接触であるので，これ以外で回転束縛自由度が 1 にならないと仮定した．

次に，Fig. 12 に示す辺一面接触の場合，式 (22) で示すように，軸 1 回りに  $\beta$  回転し，軸 2 回りに  $\alpha$  回転する姿勢変化後も接触状態を維持することができる．ただし， $\mathbf{n}$  は法線の向きを， $\mathbf{l}$  は辺の向きを表す．軸 2 の向きは軸 1 に関する回転の影響を受けない．

$$R(\mathbf{n}, \alpha)R(\mathbf{l}, \beta) \quad (22)$$

また，Fig. 13 に示す面一面接触の場合は，式 (23) で示すように，軸 I 回りに  $\beta$  回転し，軸 II 回りに  $\alpha$  回転する姿勢変化後も接触状態を維持することができる．軸 II の向きは軸 I に関する回転の影響を受けない．

$$R(\mathbf{l}, \alpha)R(\mathbf{n}, \beta) \quad (23)$$

回転束縛自由度が 0 から 1 の場合については，式 (20) の

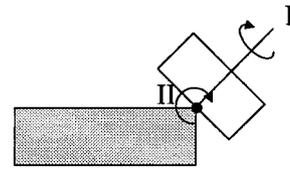


Fig. 13 Face-edge contact

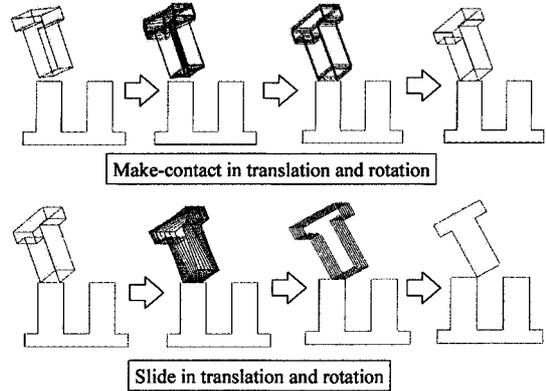


Fig. 14 Optimal trajectory

$R(\mathbf{m}, \beta)$  の代わりに式 (22)，(23) を用いることにより求めることができる．回転束縛自由度が 1 から 3 の場合についても  $\Theta_s, \Theta_e$  の差を，式 (22)，(23) で表すことにより求めることができ，いずれの場合でも問題なく最適軌道を生成することができる．

#### 4.2 回転変位の Case 3. で線形部分空間にならない場合の対処法

ここでも回転束縛自由度が 1 になるのは，辺一面接触か面一面接触のみであると仮定した．

Fig. 12 に示す辺一面接触の場合，軸 2 回りに適当に回転させることによって，式 (15) の解が式 (14) の解の線形部分空間になるようにできる．その状態で，Case 3. の方法を適用することにより，計算することができる．たいていの場合軸 2 の向きは  $\mathbf{a}_2$  に等しいので， $\mathbf{a}_1$  を式 (15) の解の基底とし， $\mathbf{a}_2$  を面の法線としたとして Case 3. を適用することにより，最適姿勢変位を求めることができる．

同様に Fig. 13 に示す面一面接触の場合でも，同様に軸 II 回りに回転させることによって，式 (15) の解が式 (14) の解の線形部分空間になるようにできる．たいていの場合，軸 II の向きは  $\mathbf{a}_1$  に等しいので， $\mathbf{a}_1$  を式 (15) の解の基底とし， $\mathbf{a}_2$  を面の法線としたとして Case 3. を適用することにより，最適姿勢変位を求めることができる．

#### 4.3 最適軌道生成実験 2

本章で提案した解決法を実装し，今度はpegとホールとの奥行き方向を合わせていない状態から最適軌道を生成をした．特に本手法とかかわる初め二つの状態遷移 (Fig. 8 (1), (2) の部分に対応する) を実現する最適軌道を Fig. 14 に示す．

今回は，位置と姿勢の両方を修正する必要が生じたため，最初に姿勢を修正して，次に位置を修正するという方法で修正を

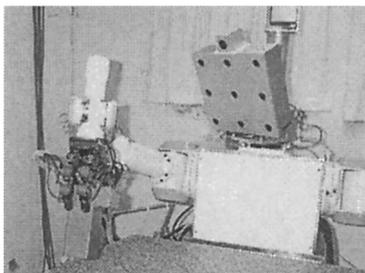


Fig. 15 Test bed

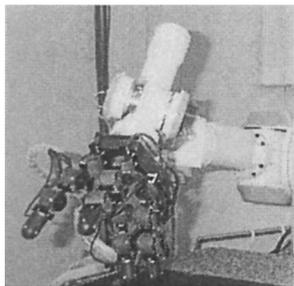


Fig. 16 Robot arm

行った<sup>†</sup>. これにより, 回転束縛自由度が1から変化する際にも, 問題なく最適軌道が生成がされていることが分かる.

### 5. ロボットへの実装

前章まで, サブスキル表現から最適軌道を生成する方法について述べた. 各物体の位置が正確に分かっており, ロボットマニピュレータが誤差なく動くという理想的な環境下では, 先に得られた最適軌道をそのままロボットアームの指令として与えることにより組み立て作業を実行することができる. しかし, 現実には種々の誤差が生じるためセンサフィードバックが不可欠となる. 本章では, ロボットへの実装に対する試みについて述べる. Fig. 15 は, 実際に用いたロボット [21] である. 本ロボットは, 運動系として7自由度ロボットアーム (三菱重工製 PA-10) と多指手を備えており, 視覚系として9眼の実時間ステレオカメラ装置を搭載している. Fig. 16 に示す多指手は, 3自由度 (安川電機製サーボモータ) の指4本から構成され, 各指の先端にはフォーストルクセンサを配している.

本章では, 初めに我々が提案するサブスキルによる組み立て作業記述の方法について述べ, 次にその記述からロボット動作コマンドへと変換するための方法について述べる.

#### 5.1 サブスキルを用いた組み立て作業記述

前述の通り, 我々は「スキルに基づくマニピュレーションシステム」[16]の考え方にに基づき, サブスキルの連続で組み立て作業を記述する方法を提案している [11]. 具体的には, 1. 接触状態を物体のとりうる微小変位に基づいて分類し, 2. その遷移に伴う微小変位の変化から対応するサブスキルを選択する, という2ステップからなる. ここでは, ロボットへの実装に重要な2. について具体的に記す.

<sup>†</sup>式 (21) の冗長な解をうまく解くことにより, 位置, 姿勢を同時に合わせる軌道を生成することは可能であるが, ここでは扱わない.

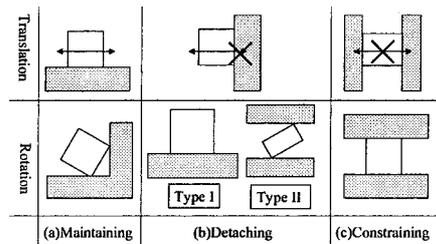


Fig. 17 Maintaining, detaching, and constraining degree of freedoms(DOFs)

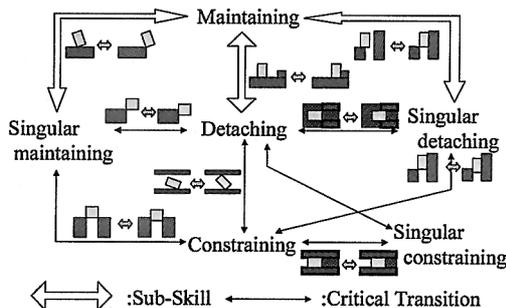


Fig. 18 Possible transitions

微小変位は, その性質の違いから維持, 離脱, 拘束, 特異維持, 特異離脱, 特異拘束の6種類に分類することができる [11]. Fig. 17 にその中でも特に重要な維持, 離脱, 拘束自由度を示す.

- 維持自由度 接触状態を維持する運動の自由度
  - 離脱自由度 接触要素数が減少する方向に接触状態を変化させる運動の自由度
  - 拘束自由度 接触により消失した運動の自由度
- 実際にこれらの自由度は文献 [22] や文献 [23] 等の手法を用いることにより求めることができる.

組み立て作業の進行に伴い, 接触状態, および運動自由度は刻々と変化していくが, 我々はある一つの自由度に対する運動自由度に注目したとき Fig. 18 に示す 20 種類の遷移しか存在しないことを示した [11].

これらの 20 種類の遷移は, その特徴により 1. 維持 ↔ 離脱, 維持 ↔ 特異維持, 維持 ↔ 特異離脱の六つの遷移と, 2. それ以外の遷移の 2 種類にさらに分けることができる.

前者は物体の運動方向と遷移する自由度の方向が等しいという特徴を持っている. これらの遷移を具体的に調べることにより, 組み立て作業に必要な不可欠な動作要素であるサブスキルを定義した (Fig. 19 参照).

後者は物体の運動方向と遷移する自由度の方向が直交しているという特徴を持っている. これらの遷移を具体的に調べることにより, その方向の誤差が組み立て作業の進行に深刻な影響を与える遷移である critical transition を定義した. Fig. 20 に, critical transition である, 特異維持 → 拘束, 特異維持 → 離脱の二つの遷移を示す.

#### 5.2 制御

実環境下でロボットに作業をさせる際には, 物体の配置位置のずれ等の誤差が発生してしまうため, その誤差を吸収するた

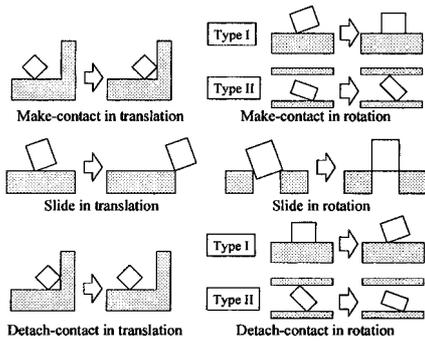


Fig. 19 Sub-skills

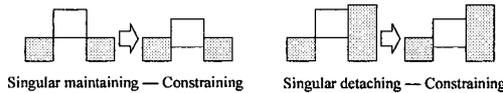


Fig. 20 Critical transitions

めの機構が必要となる。

本実装では、簡単のため以下のような仮定を設けた：

- (1) ロボットの指は物体を把持するのみで、操作は行わない。
- (2) 慣性の効果が無視できる、静的な運動で近似できる程度の速度で動かす。
- (3) 指関節には適当なコンプライアンスが設定されている。
- (4) 観察で得られた以外の接触状態、および遷移は発生しない。
- (5) 物体の把持位置は既知であり、作業中は把持位置が大きくずれることはない。

サブスキルを用いた組み立て作業の再現は以下のような手順を経て行われる。まず、視覚により環境物体の位置姿勢を獲得する。これにより、把持物体との相対位置関係を知ることができる。次に、各サブスキルを終了条件が満たされるまで実行し、終了条件を満たした段階で次のサブスキルを呼び出すことにより、組み立て作業を遂行していく。

各サブスキルは、簡単のため十分短い周期で位置制御指令を発行することによって実装した。一つのサブスキルは二つの直接遷移可能な接触状態間の遷移を引き起こすものであり、最適軌道  $q_d(s)$  は前述の方法で計算されているものとする。実際に  $t$  ステップ目に与えた位置姿勢  $q(t)$  は、式 (24) で計算した。ただし、 $f, f_d$  はフォーストルクセンサの現在値および目標値を、 $K$  はゲインパラメータを表し、 $\Delta s$  は指令値  $q(t)$  が条件 (2) を満たす程度の大きさになるように調整されているものとする。

$$q(t) = q(t-1) + (q_d(\Delta s \cdot t) - q_d(\Delta s \cdot (t-1))) + K \cdot (f - f_d) \tag{24}$$

本実装では、指先にあるフォーストルクセンサを用いて把持物体にかかるフォーストルク値を推定したが、把持位置の誤推定等によりトルクに関して正確な推定を行えなかったため、トルクによる制御は行えなかった。

5.3 並進変位誤差修正のためのパラメータの決定指針

本節では、式 (24) のパラメータ  $K, f_d$  の並進変位に関する

<sup>†</sup>Kitagaki らは、疑似接触点の変化に基づいて [24]、下倉と武藤は、軌道の変化に基づいて、接触状態遷移を判定する方法を提案している [25]。

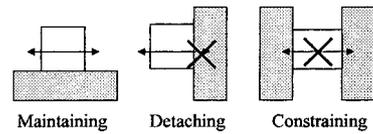


Fig. 21 Three types of translation DOFs

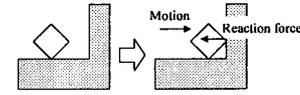


Fig. 22 Make-contact sub-skill

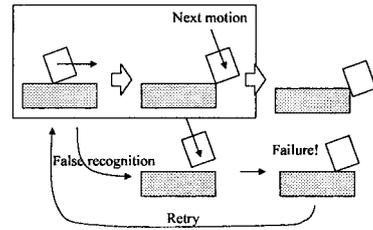


Fig. 23 Slide sub-skill

部分を決定する方法について述べる。

サブスキル実行時には、並進自由度に関して Fig. 21 に示す維持、離脱、拘束の三つの自由度が現れる。維持自由度では、多少の誤差がサブスキルの実行に影響を与えることはないので、誤差を修正する必要はない。残る二つに関しては、フォースフィードバックを施すことにより実行時の誤差を吸収する必要がある。離脱自由度では、接触を維持しながら動かすために、物体の可動範囲と反対向きに微小な力で押しやることによって誤差を修正しながら動かすことが可能となる。拘束自由度では、その自由度に関して誤差は生じないが、過大な拘束力が生じないようにするため、力が発生しないように制御する。まとめると、以下のようなになる。

- 維持自由度ならば何もしない
- 離脱自由度ならば可動方向と反対に微小な押し付け力を発生させる
- 拘束自由度ならば力を発生させないようにする

5.4 終了条件

本節では、サブスキルの終了を判定する方法を述べる。ここでは簡単のため、センサ値を用いて検出するように実装した<sup>†</sup>。サブスキルは大きく分けてつき当て、つき放し、すべらし動作の三つに分けられるが、大きな違いはその動作の終了状態を検出するための方法にある。

5.4.1 つき当て、つき放し動作の終了条件

Fig. 22 に示すようなつき当て動作では、その性質から動作終了時に運動方向と反対向きの力（トルク）が発生する。そこで、この力（トルク）を力センサを用いて検出することによりサブスキルの終了を判定することができる。つき放し動作ではその逆が成り立つ。

5.4.2 すべらし動作の終了条件

Fig. 23 に示されるようなすべらし動作では、動作終了時にある接触が特異な接触になってしまう。その際、特異な接触

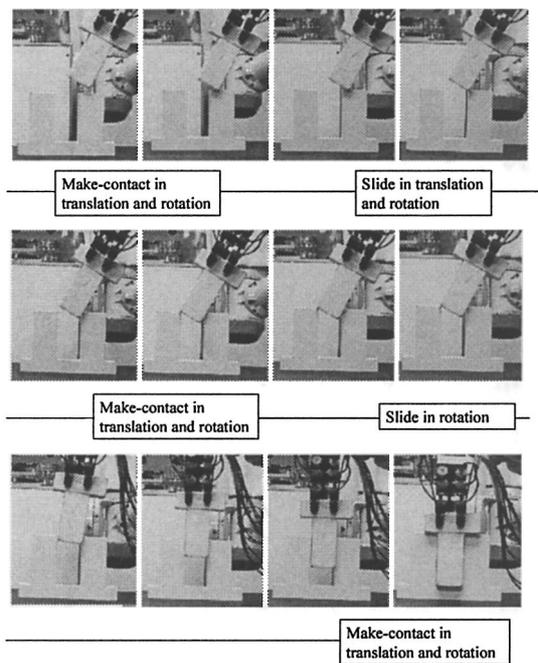


Fig. 24 Robot execution

に変わってしまう接触要素に関係する面の法線方向（辺—辺接触の場合は両辺に垂直な方向）の力が急激に減少する．これを用いてサブスキルの終了を判定することができる．ただし，その力の減少が終了を示すのか接触を維持しながら動かすことができなかったことを示すのかを区別することは難しい．

そこで，終了の可能性が生じた段階でビジュアルフィードバックにより判定する方法が考えられるが，より簡単な方法として，いったん終了したとして次のサブスキルを実行するという方法がある．なぜならば，すべらし動作が完了していない状態では，次のサブスキルは実行できないことが多いからである．

### 5.5 実験

条件 (4) を実現するために，1. 拘束度を一度に一つずつに増やしていく必要があること，2. 三次元のベグインサーションでは穴の上を通りすぎることによって，観察で得られた以外の接触状態が発生する可能性が高い，という理由から，二次元的なベグインサーションを実験対象として選んだ．ただし，運動生成は三次元空間で行った．実際にロボットが同じ組み立て作業を再現をする様子を Fig. 24 に示す．

本実験では， $f_d$  および  $K$  の値の大きさは，作業がうまく遂行されるように適当に人間が調整した．Fig. 25 は接触のない状態から，つき当て動作を行い，そしてすべらし動作を行っている際 (Fig. 24 の上段部分) の，上下方向にかかる力の時間変化を表している．この実験での目標力の値は 50 [gf] である．

5.2 [s] 付近で急激な反発力を検出してつき当て動作が終了したとみなし，すべらし動作が呼び出されている．その後徐々にではあるが目標力の値 50 [gf] に収束している．15.4 [s] 付近では，辺—面接触が消失したことによる力の減少を検出して，サ

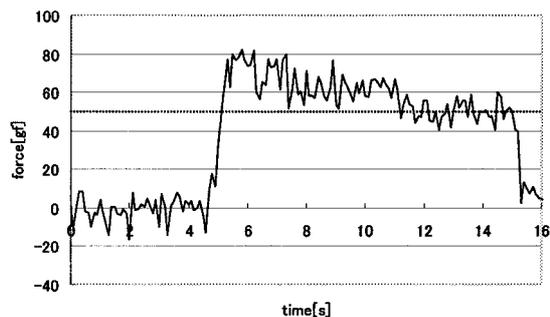


Fig. 25 Force sensor data from the 1st subskill to the 3rd subskill

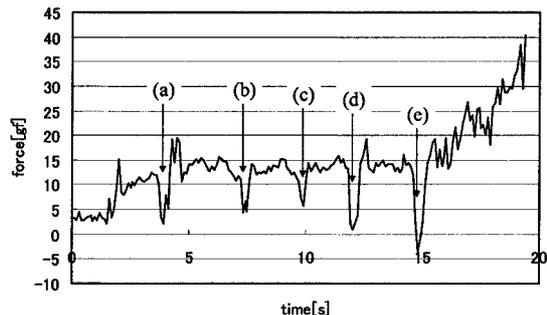


Fig. 26 Force sensor data in "slide in rotation" subskill

ブスキルが終了したとみなし，次のサブスキルを実行している．この遷移は critical transition でない，つまり少々の誤差が組み立て作業の進行に影響を与えない遷移であるので，問題なく次のサブスキルが実行されている．

Fig. 26 は回転すべらし動作からつき当て動作へ遷移する際 (Fig. 24 の中段から下段にかけての部分) の上下方向にかかる力の時間変化を表している．

この遷移は critical transition である，つまり実行時誤差が組み立て作業の進行に大きな影響を与える遷移である．図中 (a)～(d) において，上下方向の力の減少からすべらし動作が終了したとみなし，つき当て動作の呼び出しが行われているが，正確にベグを挿入可能な姿勢をとらずに次のサブスキルを呼び出した結果，直後に反発力が発生している．そこで，すべらし動作の終了判定の失敗<sup>†</sup>とみなし，再びすべらし動作を呼び出している．図中 (e) においては，同様につき当て動作を呼び出し，その結果すべらし動作が正しく終了したと判定し，続けてつき当て動作を実行しているが ( $f_d$  の値を大きめにとってある)，まだ完全に姿勢を合わせきれていないため，挿入が進むにつれて，過大な拘束力が発生している．本実験ではトルクによるフィードバックが行えなかったためこのような問題が発生したと考えられる．

## 6. 検 討

### 6.1 スキルに基づくマニピュレーションシステム

本手法は「スキルに基づくマニピュレーションシステム」[16] [25] [26] の考えに基づいている．ここでは過去の研究と本研究の関連について述べる．末広と高瀬は，平面運動に限定したうえで，接触状態を接触点の数に基づいて分類し，その間の

<sup>†</sup>1 [s] 未満で次のサブスキルが終了した場合，終了判定の失敗と判断している．

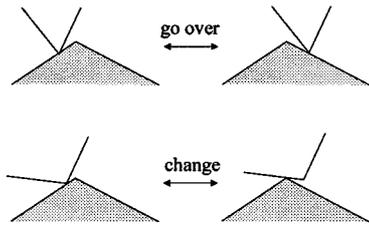


Fig. 27 “Go over” skill and “change” skill [16]

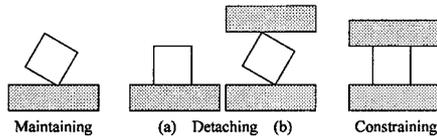


Fig. 28 Three types of rotation DOFs

遷移からスキルを導出している [16]。そこでは、以下に示す七つのスキルが提案されている。ただし、括弧内は我々が接触状態遷移モデルから導き出したサブスキル [11] のうち対応するものを示す。

- つき当て (接触がない状態で並進つき当て)
- すべらし当て (接触を維持しながら並進つき当て)
- 辺あわせ動作 (回転つき当て Type I)
- 回転つき当て (回転つき当て Type II)
- 乗り換え (並進すべらし → 並進つき当て (すべらし) もしくは、回転すべらし → 回転つき当て (すべらし))
- 乗り越え (並進すべらし → 並進つき当て (すべらし) もしくは、回転すべらし → 回転つき当て (すべらし))
- はめ遣い (回転すべらし → 並進つき当て (すべらし))

スキルとサブスキルにおける違いは以下のようなものが挙げられる：

- すべらしサブスキル  $+\alpha$  で一つのスキルとなっている。
- サブスキルには乗り換え (Fig. 27 の “change”) と乗り越え (Fig. 27 の “go over”) の区別がない。
- 並進すべらしサブスキルから回転つき当て (すべらし) サブスキルに対応するスキルが存在しない。

しかし、上に挙げた違いを除けばたいへん似かよっており、過去の研究に対する適用も自然に行うことができると考えられる。

### 6.2 姿勢誤差修正のためのパラメータの決定

並進自由度の場合と同様に、回転自由度に関しても Fig. 28 に示す維持、離脱、拘束の三つの自由度が現れる。並進自由度の場合と同様にトルクによるフィードバックを適用すべきなのは、離脱、拘束自由度に対してである。拘束自由度に関しては、並進自由度の場合と同様に過大な拘束力の問題からトルクが発生しないことが望ましい。

Fig. 28 に示したように、離脱自由度には大きく分けて 2 種類存在する。一つ目は両方向に回転することができる離脱自由度 (Fig. 28 (a) 参照) であり、もう一つは一方向への回転しかできない離脱自由度 (Fig. 28 (b) 参照) である。後者の場合は、並進自由度の場合と同様に、回転可能な方向と逆向きに微小なトルクを発生させることにより誤差を吸収することが可能となる。前者の場合は、どちらか一方の接触が離れてしまったときに物

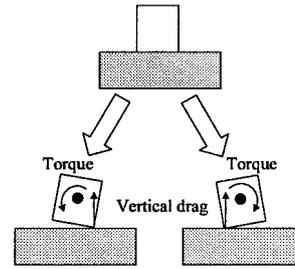


Fig. 29 Detaching DOF around where the object can rotate both clockwise and counter-clockwise

体にはそれに伴ったトルクが発生するので、物体にトルクを発生させないような制御を行うこととする (Fig. 29 参照)。

さらに、回転自由度に関しては、コンプライアンス中心をどのように選択するかという問題が残されている。清水と小菅は、コンプライアンス中心の存在領域を線形不等式を用いて定式化し求める方法を提案しており [27]、本手法に適用することは可能である。

## 7. ま と め

本論文では、視覚により得られた組み立て作業理解のための抽象表現であるサブスキルから、把持物体の最適軌道を生成し、制御指針を得る方法について述べた。まず与えられた接触状態遷移を実現するための最適な軌道を生成する方法について述べた。本来最適な軌道を生成するという問題は非線形冗長連立方程式を解くことと同様の難易度を持つが、回転軸の向きは回転運動中一定であるという制約を加えることによって、線形方程式を解くことにより、解を導くことができることを示した。

次に、ロボットへの実装について述べた。制御の指針が、定義した運動自由度の種類によって一義的に決定できることについて述べ、認識系と実行系の親和性の良さを示した。これは「見まねによる行動獲得」における動作プリミティブの表現法として適当であることを示している。また、過去に提案されている「スキルに基づくマニピュレーションシステム」との関係について述べ、過去の研究の本手法への適用が自然に行われることを示した。

最後に、ロボットによる検証を試みた結果、物体の配置等の誤差を含む環境にある程度柔軟に対応できることを示した。

## 8. 今後の課題

実装に関しては、まだまだ多くの課題を残しているといわざるを得ない。以下にその課題について述べる。

まず、観察で得られた以外の接触状態、および遷移が発生しないという仮定は、例えば直接ペグを挿入する場合では成り立たない。その問題を解決するためには、1. 実行時の誤差の結果どのような接触状態に遷移し得るのか、2. 実際に実行時にどの接触状態に遷移したのかをどのように判定すればよいのか、3. 元々意図した接触状態に戻るためにはどのような動作を適用すればよいのか、を考察する必要がある。

1. に関しては、ある接触状態の近傍の接触状態を探索する方法が提案されている [28] [29]。2. については探り動作を行うこ

とにより、接触状態を推定する方法が提案されており [30], 1. により誤った接触状態候補が絞り出されているならば, さらに効率的な探り動作を生成することができると考えられる. 3. に関しては, 2. の問題が解決すれば, 意図した接触状態に戻るためのサブスキルを導出することは可能であると考えられる. よりロバストな実行のためには, それらの実装は必要不可欠であり, 今後の課題となる.

また, 本論文では制御のためのパラメータの自動獲得法については扱わなかったが, 例えば松岡らは, あらかじめ作業の失敗を検出する方法を決定しておき, 失敗した際には, その失敗を解消する方向へパラメータを修正することにより自動的に適切なパラメータを獲得する方法を提案している [31]. どのようにして失敗検出の方法を自動的に獲得するのかといった問題点はあるものの, 実行時の誤差によって発生する接触状態を推定することにより, 自動化することができると考えられる.

謝辞 なお本研究は, 文科省科研費補助金特定領域研究 (C) 課題番号 14019027 および, 科学技術振興事業団・池内 CREST プロジェクトの補助を受けている.

### 参考文献

- [1] K. Ikeuchi and T. Suehiro: "Toward an assembly plan from observation part i: Task recognition with polyhedral objects," *IEEE Trans. on Robotics and Automation*, vol.10, no.3, pp.368-385, 1994.
- [2] S.B. Kang and K. Ikeuchi: "Toward automatic robot instruction from perception - recognizing a grasp from observation," *IEEE Trans. on Robotics and Automation*, vol.9, no.4, pp.432-443, 1993.
- [3] 津田雅之, 高橋友一, 小方博之: "実演教示による組立作業モデルの生成", *日本ロボット学会誌*, vol.18, no.4, pp.535-544, 2000.
- [4] 音田弘, 小笠原司, 比留川博久, 北垣高成, 中村見, 築根秀男: "実環境行動計画機能に基づく作業ロボットシステム", *日本ロボット学会誌*, vol.18, no.7, pp.979-994, 2000.
- [5] Y. Kuniyoshi, M. Inaba and H. Inoue: "Learning by watching: Extracting reusable task knowledge from visual observation of human performance," *IEEE Trans. on Robotics and Automation*, vol.10, no.6, pp.799-822, 1994.
- [6] C.G. Atkeson and S. Schaal: "Robot learning from demonstration," *Int. Conf. on Machine Learning*, vol.2, no.12, pp.12-20, 1997.
- [7] H. Miyamoto, S. Schaal, F. Gandolfo, H. Gomi, Y. Koike, R. Osu, E. Nakano, Y. Wada and M. Kawato: "A kendama learning robot based on bi-directional theory," *Neural Networks*, vol.9, no.8, pp.1281-1302, 1996.
- [8] S. Schaal: "Is imitation learning the route to humanoid robots?," *Trends in Cognitive Sciences*, vol.3, pp.233-242, 1999.
- [9] A.J. Ijspeert, J. Nakanishi and S. Schaal: "Movement imitation with nonlinear dynamical systems in humanoid robots," *IEEE Int. Conf. on Robotics and Automation*, pp.1398-1403, 2002.
- [10] Y. Maeda, N. Ishido, H. Kikuchi and T. Arai: "Teaching of grasp/graspless manipulation for industrial robots by human demonstration," *IEEE Int. Conf. on Intelligent Robots and Systems*, pp.1523-1528, 2002.
- [11] J. Takamatsu, H. Tominaga, K. Ogawara, H. Kimura and K. Ikeuchi: "Extracting manipulation skills from observation," *IEEE Int. Conf. on Intelligent Robots and Systems*, vol.1, pp.584-589, 2000.
- [12] B.J. McCarragher and H. Asada: "A discrete event approach to the control of robotic assembly tasks," *IEEE Int. Conf. on Robotics and Automation*, pp.331-336, 1997.
- [13] H. Hirukawa: "On motion planning of polyhedra in contact," *WAFR*, 1996.
- [14] L.E. Kavraki, P. Svestka, J. Latombe and M.H. Overmars: "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. on Robotics and Automation*, vol.12, no.4, pp.566-580, 1996.
- [15] X. Ji and J. Xiao: "Planning motion compliant to complex contact states," *IEEE Int. Conf. on Robotics and Automation*, pp.1512-1517, 2001.
- [16] 末広尚士, 高瀬國克: "スキルに基づくマニピュレーションシステム", *日本ロボット学会誌*, vol.8, no.5, pp.551-562, 1990.
- [17] 比留川博久, 松井俊浩, 高瀬國克: "多面体間の接触による拘束条件を幾何モデルから導出する一般的なアルゴリズム", *日本ロボット学会誌*, vol.9, no.4, pp.415-426, 1991.
- [18] M.S. Ohwovoriole and B. Roth: "An extension of screw theory," *Journal of Mechanical Design*, vol.103, pp.725-735, 1981.
- [19] 比留川博久: "組立作業過程における拘束の表現と解析", *日本ロボット学会誌*, vol.11, no.2, pp.192-200, 1993.
- [20] J. Takamatsu, H. Kimura and K. Ikeuchi: "Improved screw theory using second order terms," *IEEE Int. Conf. on Intelligent Robots and Systems*, pp.1614-1618, 2002.
- [21] K. Ogawara, J. Takamatsu, S. Iba, T. Tanuki, Y. Sato, A. Saegusa, H. Kimura and K. Ikeuchi: "Acquiring hand-action models in task and behavior levels by a learning robot through observing human demonstrations," *The First IEEE-RAS Int. Conf. on Humanoid Robots*, 2000.
- [22] 比留川博久, 松井俊浩, 高瀬國克: "多面体間の接触による拘束条件の高速解法とその離脱動作計画への応用", *日本ロボット学会誌*, vol.9, no.7, pp.841-848, 1991.
- [23] 平井慎一: "凸多面体理論に基づくマニピュレーションの運動学・静力学とそのマニピュレーション計画問題への応用", *日本ロボット学会誌*, vol.17, no.1, pp.68-83, 1999.
- [24] K. Kitagaki, T. Suehiro and T. Ogasawara: "Monitoring of a pseudo contact point for fine manipulation," *IEEE Int. Conf. on Intelligent Robots and Systems*, pp.757-762, 1996.
- [25] 下倉健一朗, 武藤伸洋: "接触状態変化検出機能を有するマニピュレーションシステムに関する検討", *日本ロボット学会誌*, vol.12, no.6, pp.837-845, 1994.
- [26] 下倉健一朗, 武藤伸洋: "Assembly task-oriented manipulation system (ATOMS) の開発", *日本ロボット学会誌*, vol.13, no.8, pp.1190-1198, 1995.
- [27] 清水昌幸, 小菅一弘: "Structured compliance を用いた平面組付作業", *日本ロボット学会誌*, vol.20, no.8, pp.852-859, 2002.
- [28] 平井慎一, 浅田春比古, 得丸英勝: "組立作業における物体の接触状態遷移に関する運動学的解析とそのネットワーク表現の自動生成", *計測自動制御学会論文集*, vol.24, no.4, pp.406-413, 1988.
- [29] J. Xiao and X. Ji: "A divide-and-merge approach to automatic generation of contact states and planning of contact motion," *IEEE Int. Conf. on Robotics and Automation*, pp.750-756, 2000.
- [30] 余永, 吉川恒夫, 辻尾昇三: "対象物と環境の接触位置を推定する探り動作の自動生成", *日本ロボット学会誌*, vol.16, no.3, pp.417-424, 1998.
- [31] 松岡毅, 長谷川勉, 本田久平, 桐木利弘: "作業状態観測と評価に基づく多関節多指ハンド物体操作システム", *日本ロボット学会誌*, vol.17, no.5, pp.696-703, 1999.



高松 淳 (Jun Takamatsu)

1974年8月20日生。1999年東京大学理学部情報科学科卒業。2004年東京大学大学院情報理工学系研究科コンピュータ科学専攻博士課程修了。情報理工学博士。知能ロボットの動作獲得に関する研究に従事。  
(日本ロボット学会正会員)



小川原光一 (Koichi Ogawara)

1975年2月5日生。1997年東京大学工学部機械情報工学科卒業。2002年東京大学大学院工学系研究科電子情報工学専攻博士課程修了。工学博士。2002年より科学技術振興機構博士研究員。知能ロボットの動作獲得に関する研究に従事。IEEEの会員。  
(日本ロボット学会正会員)



木村 浩 (Hiroshi Kimura)

1961年2月5日生。1983年東京大学工学部機械工学科卒業。1988年東京大学大学院工学系研究科機械工学専攻博士課程修了。工学博士。東北大学工学部助手、電気通信大学講師を経て、現在、電気通信大学情報システム学研究所助教授。1995～1996年カーネギーメロン大学客員研究員。脚型ロボット、知能ロボットの研究に従事。1989, 2003年度日本ロボット学会論文賞受賞。日本機械学会, IEEE等の会員。  
(日本ロボット学会正会員)



池内克史 (Katsushi Ikeuchi)

1949年5月29日生。1973年京都大学工学部機械工学科卒業。1978年東京大学大学院工学系研究科情報工学専攻博士課程修了。MIT人工知能研究所、電総研、CMU計算機科学科を経て、1996年より東京大学生産技術研究所教授。2000年より東京大学大学院情報学環教授兼担。人間の視覚機能、明るさ解析、物体認識、人間による組み立て作業の自動認識などの研究に従事。工学博士。D.Marr賞 (ICCV:1990年), IEEE優秀論文賞 (CVPR:1991年), 最多引用論文賞 (AI Journal:1992年)等受賞。IEEE Fellow。  
(日本ロボット学会正会員)