教示資産の活用と作業意図の反映のための経路生成法

小 方 博 之* 高 橋 友 一*

Path Planning Based on Task Knowledge and User's Intention

Hiroyuki Ogata* and Tomoichi Takahashi*

This paper introduces a method to extract a task knowledge from an example shown by an operator or generated by a planner, and to apply the knowledge to plan a path in similar environments. Our aim is to make efficient use of learning task resource and to plan a path in complex environments easily. Our method is based on the A* algorithm. We developed a technique to generate a suboptimal path with quite less amount of search nodes than the traditional A* algorithm, and to make a heuristic function that includes task knowledge. Examples are shown to verify the effect of our method.

Key Words: Path Planning, Teaching, Task Knowledge, User's Intention, Replanning

1. は じ め に

本論文は教示などで得た作業動作を手掛かりに、類似の作業を実行するための経路を生成する手法を提案する。これは、教示資産の活用と、設計者や教示者の意図を作業に反映させるという二つのことを目的にしている。

ロボットに作業を実行させる場合、同様な作業でも部品の寸法やデザインの変更、作業環境の変更などがあると再教示の必要が生じる。しかし、作業の内容が本質的には変わらないのに、教示を最初からやりなおすのは非効率的である。したがって、以前に同じような作業を行った経験があれば、その知識を利用して経路生成の効率化を図るのは自然なことといえる。実際、それを目的にオフライン教示ではパラメトリック教示という手法が開発されているが、相似な環境で相似な部品を組み立てる作業に限られる。

また、組立作業の場合、人間は部品をどのように組み立てるか考えながら設計する。しかし、経路計画時にはこのような考えは情報として活用されず、以下の問題が生じている。

● 実用的な経路生成手法が出てこない

スタートとゴールを与えると自動的に経路を計算する経路計画の研究は盛んに行われているが[4]、部品数が多い場合や、次元数が大きくなる場合には実用的な時間で経路を生成できるものはほとんどない。経路計画の困難さは古くから指摘されており[13]、あらかじめ存在する情報を活用して計画の困難さに対処する手法を開発する必要がある。

● ユーザの意図が伝わらない

経路計画を行う場合には何らかのヒューリスティクスが用いられるが、そのヒューリスティクスは必ずしもユーザの意図する経路を生成するものではない。また、ユーザの求める経路を生成するような経路計画用のヒューリスティクスの作成方法は知られていない。

何らかの方法で設計者や教示者の作業意図をプランニングに取り込める形に情報化し経路生成に利用できれば、これらの問題は解決できると考えられる。

組立作業を教示から情報化する試みとしては、物体の最終位置の關係に着目したもの[8]、物体の組立プロセスの局所的な幾何關係の変化に着目したもの[5]がある。また、設計図から情報化を行うもの[1]もある。しかし、いずれも獲得された情報を組立経路生成に利用する本格的な手法は論じていない。

筆者らは物体の組立プロセスの大域的な幾何關係の変化に着目して組立作業を教示から情報化する研究を進めてきた[10]。本論文では教示から得られた情報を異なる環境の経路生成に利用する手法を提案する。本手法は自由空間にA*アルゴリズムを適用して経路を生成する。A*アルゴリズムを自由空間の経路生成に利用した研究には論文[14]などがあるが、本論文は、A*アルゴリズムに用いるヒューリスティック関数を教示などで得られた情報から作成していることと、探索方向が大体判明している場合にさらに探索空間を小さくするため高速に準最適解を発見する新手法を提案していることに特徴がある。

第2章と第3章では例題から幾何關係の変化に着目して作業意図を抽出し汎化する手法を簡単に述べる。第4章ではそれを類似の環境に転用する方法を論じる。そして、第5章ではそれを用いて異なる環境で経路を生成する方法を示し、第6章では

原稿受付 1994年3月17日

*日本電信電話株式会社

*Nippon Telegraph and Telephone Co.

計算機上での評価を掲げる。

2. 作業意図の抽出

教示や計画の作業意図を抽出するために、組立経路から抽象的な作業記述を作成する。組立経路自体は環境と操作物体の関係を明確に含んでいないので、そのまま利用できない。筆者らは並進操作で行う組立において、組立経路から環境に対する幾何的条件の遷移に着目して、抽象的な作業記述を作成する方法を提案した[10]。

作業記述は作業状態の列で表現される。作業状態は以下に説明するように、C-space (configuration space) [9] における幾何拘束の集合で表す。

C-space とは組立作業時に操作する物体の可能な位置・姿勢を表現した空間である。C-space の特徴は操作する物体を点とみなせることにある。実環境での他の物体は C-space では C-object と呼ばれる物体に変換される。

環境中の物体が多面体で、操作物体が並進移動のみを行う場合は C-object も多面体になる。C-object の各面は C-function と呼ばれる関数 $f_{ij}(\mathbf{x})$ を用いて

$$f_{ij}(\mathbf{x}) = 0 \quad (1)$$

で表される。ここで、

$$f_{ij}(\mathbf{x}) \stackrel{\text{def}}{=} \langle \mathbf{n}, \mathbf{x} \rangle - \langle \mathbf{n}, \mathbf{a}_i + \mathbf{b}_j \rangle \quad (2)$$

である。 i と j は C-object の頂点、稜および面に通して付けられた番号で、 i と j の組み合わせとしては頂点と面、稜と稜、面と頂点の3種類のみを許す。 $\mathbf{n}$ は C-object の該当する面の外向き法線である。 $\mathbf{a}_i$ と $\mathbf{b}_j$ はそれぞれ i , j の頂点、稜または面から規定されるベクトルである。

i と j の組合せによっては存在しない面もあるが、Donald の applicability constraint function を用いることで判別できる[3]。

式(1)を C-object の面ではなく、1枚の平面とみなした場合、その平面を用いて C-space を二つの半空間に分割できる。この半空間は、C-function を用いて

$$f_{ij}(\mathbf{x}) \geq 0 \text{ or } \leq 0 \quad (3)$$

と表される。この半空間分割を繰り返すことで C-space を分割して得られた多面体領域を作業状態とする (Fig. 1)。よって、作業状態は式(3)の不等式拘束の集合で記述される。

操作物体が初期位置からどの作業状態を経て組み立てられたかを示す作業状態列で作業記述を生成する。

3. 作業記述の汎化

しかし、前章で得られた作業記述をそのまま異なる環境に適用するには問題がある。

まず、上述の拘束には作業に関係ない拘束も含まれる。例えば、Fig. 2(a)では、作業状態 S は遠方の C-object A の辺による拘束も記述として含む。 A は操作物体が作業状態 S にあるときには組立操作に影響を及ぼすとは考えにくい。それにもか

Fig. 1 Example of a task state.

Area A is represented as $\{f_1(\mathbf{x}) \leq 0, f_2(\mathbf{x}) \leq 0, f_3(\mathbf{x}) \geq 0\}$.

Fig. 2 Change of task description by a change of object's location.

かわらず、 A の位置や姿勢が大きく変化したとき (Fig. 2(b)) には作業記述に変化が生じ、作業状態 S が消滅することがある。このように、作業に無関係な拘束が作業状態の記述に含まれると不都合である。

また、環境が複雑になると、記述量が増加する。これは、別環境の作業記述に書き換えるときにマッチングの手間が増えることを意味し、拘束が多いとそれを満たす領域が得にくくなるわけで、別環境に適用するときに不都合である。

そこで、作業記述をロバストにし、記述量を小さくするために、作業に関係のない幾何拘束を判別し除去する。そのために、まず幾何拘束の分類を行う。ここでは、2通りの方法で幾何拘束を分類する。

最初の分類法は各作業状態に対する不等号の向きによる分

Fig. 3 Class of constraints.

類である。式 (3) において $f_{ij}(\mathbf{x}) \geq 0$ の形の拘束を正拘束, $f_{ij}(\mathbf{x}) \leq 0$ の形の拘束を負拘束と呼ぶ (Fig. 3(a))。

次の分類法は各作業状態を表す領域に着目した分類である。削除すると領域の形が変化する拘束を有効拘束と呼ぶ。また、変化しない拘束を無効拘束と呼ぶ (Fig. 3(b))。

作業記述の汎化は次の 4 ステップで行う。

ステップ 1 負拘束を削除する。作業を行う上で重要なのは操作物体と環境中の他の物体との衝突あるいは接触である。操作物体 A がある作業状態にあるとき、 A にとって正拘束となる面は A の方向に向いており、 A と衝突や接触が生じる可能性がある。一方、負拘束の場合は、面が A の方向に向いていないので、その可能性はない。したがって、負拘束の面は考慮しないで済む。

ステップ 2 作業状態の再編成を行う。再編成は隣接する作業状態を併合していくことで行う。 i 番目の作業状態 S_i が幾何拘束 c_{ij} で

$$S_i = \{c_{i0}, \dots, c_{im}\}$$

と表されたとする。

$$S'_i = S_i \cap S_{i+1}$$

とすると、 S'_i は S_i と S_{i+1} を含む凸多角形領域になる。 S'_i が C-object と干渉しない場合はその併合は可能なの

Fig. 4 The task will not be affected by any change of location and shape of shaded objects.

で、作業状態列の S_i と S_{i+1} を S'_i に置き換える。併合は可能な限り続ける。

ステップ 3 作業状態の拡張を行う。各作業状態の有効拘束について、それを削除した領域が C-object と干渉しなければ削除する。

ステップ 4 作業状態の無効拘束をすべて削除する。

汎化の結果、作業記述は互いに重なり合った多面体の集合になる。

上述のステップを実現するためには (1) 幾何拘束の集合が与えられた場合にそこから有効拘束を求めることと、(2) 凸体どうしの干渉を判定することが必要となるが、これらの解法は二次元と三次元いずれに対しても知られており [12]、本手法は二次元に対して適用できる。

上述の汎化の手続きによって、作業記述量の減少と環境の変化に対するロバスト性の改善が得られる [11]。まず、記述量のオーダーが $O(n^2)$ から $O(\sqrt{n})$ と大幅に減少する (n は環境の全拘束数、 d は次元)。オーダーが d 乗根なので、この記述は次元が高くなるにつれて効率はよくなるといえる。次に、本手法によって遠方の C-object からの拘束を考慮せずに済む。Fig. 4 はある環境で教示経路から作業状態を作り、汎化を行った例である。左下から右上に至る曲線は教示経路を表し、黒く塗りつぶされた領域と斜線を施した領域は物体の内部を表す。ドットを施した領域は作業記述に含まれる作業状態を表す領域で、これらは互いに重複する凸多角形である。斜線を施した物体はこれらの作業状態の表現にその拘束がまったく用いられていないものである。したがって、少なくとも斜線を施した物体が変形したり、位置・姿勢に変化が生じたり、消滅した程度では作業記述は変化せず、作業は本質的に変わらないといえる。

この抽象化によって、教示環境と類似した作業環境に対しても教示内容を適用できるようになる。

4. 作業記述の転化

上述の作業記述は物体の幾何的拘束条件で記述されている。そこで、この作業記述を新たな環境に適用するには、各拘束が新たな環境のどの拘束に対応するか知る必要が生じる。これは元の環境と新しい環境における C-object の面のうち作業記述

に用いられている拘束面の対応がとれていれば可能である．逆に，他の面は対応がとれている必要はないし，対応する面が存在しなかったり，余計な面が存在してもよい．

新たな環境で対応する部品の形状が分かっている場合，センサを用いてその物体の位置と姿勢を取得する研究は盛んに行われている [2]．これらの研究により物体の位置・姿勢が得られることを前提とすると，新しい環境で作業記述を構成する拘束を求められる．対応する拘束が求まれば，新しい環境での作業状態が一意に定まる．

環境によっては，ある作業状態の拘束をすべて満たす領域が転用先の環境に存在する場合としない場合とがある．前者の場合には，教示で得られた作業記述を変更なく用いることができるが，後者の場合にはそのまま使えない．

まず，以下のように作業環境における作業状態 S'_i を教示環境における作業状態 $S_i = \{c_{i0}, \dots, c_{im}\}$ に対応させる．

- (1) 幾何拘束 $c_{i0}, \dots, c_{im}$ を作業環境で対応する拘束に写像し，存在するものに番号を振り直したものを $c'_{i0}, \dots, c'_{in}$ とする．
- (2) 拘束 $c'_{i0}, \dots, c'_{in}$ をすべて満たす領域が作業環境に存在すれば，それを対応する作業状態 S'_i とする．領域が存在しなければ，対応する作業状態はないとみなす．

こうして作業環境における作業状態列を作成し，存在するものに番号を振り直した作業記述を $S' = S'_0, \dots, S'_p$ とする． S' が次の条件を満たす場合，作業記述は変更なく使える．

- $S' = S'_0, \dots, S'_p$ において，すべての $i = 0, \dots, p-1$ に対して S'_i と S'_{i+1} の間に共通領域が存在する．(線領域でも可)
- すべての $i = 0, \dots, p-2$ に対して， S'_i, S'_{i+1} 間の共通領域と S'_{i+1}, S'_{i+2} 間の共通領域との間に経路が存在する．
- スタートと S'_0, S'_1 間の共通領域との間に経路が存在し， S'_{p-1}, S'_p 間の共通領域とゴールとの間にも経路が存在する．

以下は，作業記述の変更を要しない例である．

- 差異が拡大，縮小，回転などのアフィンの場合 (Fig. 5(a))．
- 作業記述に幾何条件が使用されない物体が作業状態を表す領域をふさがないように移動，変形，消失した場合 (Fig. 5(b))．
- 教示環境になかった物体が作業状態を表す領域をふさがらない場所に出現した場合．
- 作業記述に幾何条件が使用される物体でも，使用されていない幾何条件が作業状態を表す領域に干渉しないように変化，消失した場合 (Fig. 5(c))．

作業記述の変更を要しない場合は作業動作は局所的な修正で済む．例えば，経路が作業状態の重複領域の重心を結ぶ折れ線で表されている場合には，その重心を再計算するだけで済む．

一方，上述の条件を満たさず，作業記述がそのまま適用できない場合でも，部分的に作業状態を対応できる場合がある (Fig. 6)．このような場合にも，対応のとれた作業状態の間に

Fig. 5 Environments that have common task descriptions with the task in Fig. 4.

Fig. 6 Environment that has partial correspondence with the task in Fig. 4.

経路計画を断片的に適用することで，何の情報もなくスタートからゴールまで計画を行うよりも効率的に経路が生成できる．

5. 経路の生成

経路の生成には，C-space を作業状態を表す領域よりも十分に細かいセルに分割してグラフ探索法を用いる．上述の作業記述からヒューリスティック関数を作成することで，教示などで得られた知識を活用し，効率的な経路生成を行える．探索アルゴリズムとしては A* アルゴリズムまたは以下に述べる探索アルゴリズムを用いる．

5.1 探索アルゴリズム

A* アルゴリズムは最小コスト探索の効率的手法として知られている．A* アルゴリズムとは探索空間をグラフで表現したときに，スタートノードからノード N までの実際のコストを表す

Fig. 7 Searched areas by changing weight of $h(N)$.

関数 $g(N)$ と、 N からゴールノードまでの推定コストを表すヒューリスティック関数（または H 関数） $h(N)$ の和

$$f(N) = g(N) + h(N) \quad (4)$$

で N を通る経路のコストを推定し、それを手掛かりに探索を行うものである [6].

A* アルゴリズムで最適解を求めるには、 $h(N)$ が

$$0 \leq h(N) \leq h^*(N) \quad (5)$$

の条件を満たす必要がある。ここで $h^*(N)$ は実際に N からゴールノードに至るときのコストの下限值で、問題が与えられたときには未知なのが普通である。 $h(N)$ が上述の条件を満たさない場合、解の最適性は保証されなくなる。

式 (4) で $g(N)$ と $h(N)$ の意味は次のように考えられる。まず、 $h(N) = 0$ とした場合の探索法は横型探索法であることが知られている。つまり、 $h(N) = 0$ とは問題に対して知識なしに探索することを意味し、探索はスタートノードを中心に全方向的に行われる。一方、 $g(N)$ を N までの実際のコストではなく 0 とし、 $h(N)$ を N からゴールノードまでの直線距離とした場合は、深さ方向の探索を優先し、ポテンシャル的な振舞いを見せる。つまり、式 (4) において、 $g(N)$ は横方向に、 $h(N)$ は深さ方向に探索を誘導するように働くと考えられる。

Fig. 7 は $g(N)$ と $h(N)$ の働きを確かめるために、ノード N からゴールまでの距離を d として、ヒューリスティック関数を

$$h(N) = ad$$

(a は係数) として a を変化させた場合の探索空間の変化を示したものである。 $a = 1$ の場合が通常の A* アルゴリズムとなる。 a が大きくなるに従って横方向の探索が抑制され探索ノード数が減少している様子が分かる。

このように、大体の探索方向が分かっている場合に $a > 1$ とすると探索量を減少させることができ、効果が期待できる。ただし、 $a > 1$ の場合は必ずしも条件式 (5) を満たさなくなるので、求められる解は準最適となる。

このことを踏まえて、ここでは次の方針で探索アルゴリズムを構築する。

- (1) 教示や計画で得られた作業意図を反映させるために、作業記述からヒューリスティック関数 $h(N)$ を作成する。
- (2) 作業記述から $h(N)$ を作成した場合は、探索方向が大体示されているので、深さ方向の探索を優先することで、探索効率を上げることができる。そこで、 $a > 1$ の係数を $h(N)$ に掛けて探索を行う。

5.2 ヒューリスティック関数の作成

作業記述の作業状態を順次通過するような経路を生成させるには、作業状態を通過していくごとに推定コスト値が減少していくようなヒューリスティック関数 $h(N)$ を作成すればよい。

そこで $h(N)$ を以下のように作成する。ある作業状態 S_i から次の作業状態 S_{i+1} に移動するまでのコスト d_i を重心間の距離で近似する。

$$d_i = \|\mathbf{c}_{i+1} - \mathbf{c}_i\|$$

($\mathbf{c}_i$ は作業状態 S_i の重心)

次に、ノード N が作業状態 S_i にあるとき、次の作業状態 S_{i+1} に移動するまでのコスト $d'(N)$ を次の値で近似する。

$$d'(N) = \|\mathbf{c}_{i+1} - \mathbf{x}(N)\|$$

ここで $\mathbf{x}(N)$ はノード N の位置ベクトルである。

これらより、 N が作業記述に含まれるいずれかの作業状態に属するとき、 $h(N)$ を次のように設定する。

$$h(N) = d'(N) + \sum_{j=i_N+1}^{n-1} d_j$$

(i_N は N のある作業状態の番号、 n は作業状態の総数)

N が作業記述に含まれる作業状態に属さないときは教示によって与えられた知識が活用できない場合と考えられる。この場合の戦略の一つとして、経路生成のための無駄な探索を避けるために、できるだけはやく教示知識の活用できる状態に持っていくことが考えられる。そこで、作業記述に含まれる各作業状態 S_i に達するまでの推定コスト e_i をそれぞれ

$$e_i(N) = \|\mathbf{c}_i - \mathbf{x}(N)\|$$

とする。 $e_i(N)$ を最小とする i が I のとき、 $h(N)$ を次のように設定する。

$$h(N) = e_I(N) + \sum_{j=I}^{n-1} d_j$$

Fig.8 Execution of learned task in different environments.

6. 評価

作業知識に基づいて $h(N)$ を作成し経路生成実験を行った。Fig.8(a) は教示環境と、そこで教示した経路である。教示環境は二つの部屋に仕切られ、右側の部屋には四つの部品がある。左側の部屋から右側の部屋に通じる入口は三つあるが、そのうち手前の二つは閉じられ、作業知識なしの場合はトラップとなる。つまり、経路生成時にここの探索が省略されたら作業知識

が活用できたとみなせる。教示内容は長方形の操作部品を、開いている一番奥の入口から搬入して丈の高い凹形の部品にはめ込むことである。教示経路を C-space で表すと Fig.8(b) のようになる。

この教示経路からは四つの作業状態で表される作業記述が生成する。各作業状態の表す領域を示したのが Fig.8(c) である(領域は互いに重複している)。

次に、この作業記述を部品の対応はとれた異なった環境に転

用する。新たな環境では部品の形状や右側の部屋の部品の位置は教示環境とは異なっている。新たな環境に作業記述を転用した結果を Fig.8(d) に示す。ここでゴールの位置も作業記述から自動的に計算される。

この環境で作業記述からヒューリスティック関数を作成し、 $a = 10$ として探索アルゴリズムを適用した様子を Fig.8(e) に示す。塗りつぶされた部分は探索されたノードである。このケースでは生成経路のノード数が 395 なのに対し、探索ノード数は 471 だった。Fig.8(f) は生成経路の実空間での表示である。

Fig.8(g) にスタートも変えた場合の探索の様子を示す。この場合は生成経路のノード数が 324 なのに対し、探索ノード数は 374 だった。Fig.8(h) は生成経路の実空間での表示である。対照のため、同一環境で $h(N) = r$ (r はノード N からゴールノードまでの直線距離) として作業知識なしで A* アルゴリズムで経路生成を行った場合の探索の様子を Fig.8(i) に示す。この場合は生成経路のノード数は 310 と優れるものの、探索ノード数は 9654 になった。

Fig.8(e), (g) を見ると探索されたノードを見るとほとんど生成経路上に集中し、探索効率のよさが確認できる。また、設定されたトラップの探索も行われず、作業知識が生かされていることが観察できる。このように本手法を用いることで、以前の作業知識を反映した効率的な経路生成が可能であることが確かめられた。

7. お わ り に

本論文では、教示や計画で与えられた経路から作業意図を抽出して類似の環境で経路を生成する手法を提案した。

本論文の目的は、教示資産や以前の経路計画で得られた資産の活用と、設計者や教示者の意図を生成経路に反映させることにある。

この手法を実現するために、本論文では (1) 作業知識がある場合に準最適解を少ない探索数で求めることを目的とした探索アルゴリズムを開発し、(2) 作業意図を反映した経路を生成す

るために、作業記述から探索アルゴリズムに用いるヒューリスティック関数を作成する方法を示し、(3) これを用いて異なる環境で同様の作業意図を持つ経路を生成できることを示した。

参 考 文 献

- [1] N. Abe, K. Ohno, T. Kitahashi and S. He: "Task Specification Using Technical Illustrations," IEEE Int. Conf. on Robotics and Automation, vol.2, pp.58-62, 1993.
- [2] D.H. Ballard and C.M. Brown: "コンピュータ・ビジョン", 日本コンピュータ協会, 1987.
- [3] B. Donald: "A Search Algorithm for Motion Planning with Six Degrees of Freedom," Artificial Intelligence 31, pp.295-353, 1987.
- [4] Y. Hwang and N. Ahuja: "Gross Motion Planning-A Survey," ACM Computing Surveys, vol.24, no.3, pp.219-191, 1992.
- [5] K. Ikeuchi, T. Suehiro, P. Tanguy and M. Wheeler: "Assembly Plan from Observation," The Robotic Institute Annual Research Review, Carnegie Melon Univ., pp.37-53, 1990.
- [6] 人工知能学会編: "人工知能ハンドブック", オーム社, 1990.
- [7] O. Khatib: "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," Int. J. of Robotics Research, vol.5, no.1, pp.90-98, 1986.
- [8] Y. Kuniyoshi, M. Inaba and H. Inoue: "Seeing, Understanding and Doing Human Task," IEEE Int. Conf. on Robotics and Automation, pp.2-9, 1992.
- [9] T. Lozano-Pérez: "Spatial Planning: A Configuration Space Approach," IEEE Transaction on Computers, vol.32, no.2, pp.108-120, 1983.
- [10] 小方博之, 高橋友一: "幾何情報に基づく作業記述法を用いて異なる作業環境で教示内容を実行するロボットシステム", 日本ロボット学会誌, vol.11, no.3, pp.444-452, 1993.
- [11] H. Ogata and T. Takahashi: "A Geometric Approach to Task Understanding and Playback-Compact and Robust Task Description for Complex Environments," IEEE Int. Conf. on Robotics and Automation, 1994.
- [12] F.P. Preparata and M.I. Shamos: "Computational Geometry: An Introduction," Springer-Verlag, 1988.
- [13] J.H. Reif: "Complexity of the Mover's Problem and Generalizations," Proc. of IEEE Symposium on Foundations of Computer Science, pp.421-427, 1979.
- [14] C.W. Warren: "Fast Path Planning Using Modified A* Method," IEEE Int. Conf. on Robotics and Automation, vol.2, pp.662-667, 1993.

小方博之 (Hiroyuki Ogata)

1965 年 10 月 17 日生。1988 年東京大学工学部精密機械工学科卒業。1990 年同大学大学院工学系研究科修士課程修了。同年、日本電信電話株式会社入社。現在、同社ヒューマンインタフェース研究所にてロボット教示システムの研究に従事。精密工学会の会員。(日本ロボット学会正会員)

高橋友一 (Tomoichi Takahashi)

1952 年 3 月 12 日生。1974 年名古屋大学理学部物理学科卒業。1976 年同大学大学院工学系研究科修士課程修了後、日本電信電話公社(現・日本電信電話株式会社)入社。現在、同社ヒューマンインタフェース研究所にて知識処理、知的インタフェースの研究に従事。工学博士。電子情報通信学会、情報処理学会などの会員。