サービスロボットの構造変化に対する適応動作の高速生成*

舟洞 佑記^{*1}, 矢野 良和^{*2}, 道木 慎二^{*3}, 大熊 繁^{*1}

Rapid Motion Generation Based on Adaptive Algorithm for Service Robots

Yuki FUNABORA^{*1}, Yoshikazu YANO, Shinji DOKI and Shigeru OKUMA^{*1} Graduate School of Engineering, Nagoya University
Furo-cho, Chikusa-ku, Nagoya city, Aichi, 464-8603, Japan

We suggest rapid motion generation methods against unexpected structure changes for service robots. Service robots for a therapy need to be compact, lightweight and inexpensive. It is considered for these robots that unexpected structure changes have occurred with users accidental operations such as falling by their mistakes. Because robot users have no knowledge and cannot repair their robots, it is desired to recover the motions autonomously without repair. On trajectory control, the original motions can be realized by recovering the positions of end-effector. Conventionally search methods needs huge amounts of explorations to obtain adaptable end-effector points on the target trajectory, and conventionally adaptable methods are not intended for structure changes especially discontinuous changes. In order to reduce the costs of exploration, we suggest motion mapping methods by which robots learn the differences of joint angles between prepared robot model and changed robot structure based on position observation of the end-effector. The mapping is executed within real time scale because new motions are estimated from the low amount of position observation data which is gathered along with a motion execution. By learning the differences iteratively, the generated new motions will converge at the adapted motions achieving the desired trajectories without unnecessary explorations. In simulation, the adapted motions are generated autonomously against discontinuous changes with low number of position observations.

Key Words : Autonomy, Robot, Learning, Adaptive Algorithm, Motion Adaptation, Structure Changes, Self Organizing

1. ま え が き

現在日本では、ロボットの応用分野の拡大が進められ⁽¹⁾、中でも、介護・福祉分野におけるロボットを利用した介在療法（ロボットセラピー）が注目されている^{(2)~(4)}。産業技術総合研究所の PARO を始め実ロボットを利用したセラピーの研究成果報告もされており^{(5)~(6)}、ロボットをなでる、あやす、抱き上げるなどのロボットとのインタラクションや、会話などのコミュニケーションに基づきセラピーが行われている。このようなインタラクションを前提とするロボットは小型かつ軽量であることが望まれる。また、今後ロボットの普及に伴う量産化により、ロボットのフレームやギアは安価・軽量化される。ユーザはインタラクション中に誤ってロボットを落下させることもあり、ロボットに故障や変形など予期せぬ機械的な構造変化を与える。一方でユーザはロボットの専門知識を持たないため迅速な修理は行えず、故障時には利用を中断せざるを得ない。可能な範囲でロボットの動作を継続させる方策が求められる。

これまでロボットの機械的な構造変化となる故障に着目し、故障診断や故障補償を行う研究^{(7)~(9)}もされてきたが、故障箇所や故障状態を事前に定義したモデルを必要とし、各々の故障状況に応じた補償法も与える必要がある。多自由度多関節を持つ次世代のサービスロボットにおいて、様々な故障による構造変化を事前に想定しておく手法の適用は困難と考えられる。それに対して Peters ら⁽¹⁰⁾は、アームロボットの手先位置と関節角をセンサから

* 原稿受付 2012 年 1 月 25 日

^{*1} 正員, 名古屋大学工学研究科電子情報システム専攻 (〒464-8603 愛知県名古屋市千種区不老町)^{*2} 愛知工業大学電気学科 (〒470-0392 愛知県豊田市八草町八千草 1247)^{*3} 名古屋大学工学研究科電子情報システム専攻

E-mail: funabora@nagoya-u.jp

取得し、ロボットが目標軌道を実現する関節角を適応的に生成する手法を提案した．ニューラルネットや学習手法により適応的に構造情報を学習するが，事前に想定できない構造変化に対してはニューラルネットの構造や強化学習の学習方略を決定できない問題や，ロボットの動作数が増加すると学習が収束しない問題がある．制御分野ではモデルを逐次同定しながら所望の軌道を実現する適応制御の手法^{(11)~(13)}が研究されてきた．Ko ら⁽¹²⁾はガウス過程に基づくパーティクルフィルタやカルマンフィルタを実現し，小型飛行船の追従予測及びモデル同定を含む移動制御を実現した．Deisenroth ら⁽¹³⁾はガウス過程に基づく独自のフィルタ GP-ADF を提案し，再帰的にフィルタ処理を行うことでノイズを含む非線形の内部状態を精度よく同定しながら制御した．これら手法は複雑で非線形な構造を持つサービスロボットへの適用も期待でき，ロボットの構造も逐次学習することが可能であるが，モデルそのものの変化は想定されておらず，特に場合分けが必要となるモデルの学習は困難となる．例えば，落下によるギア破損でギアに空転範囲ができると，破損部を境にロボットのモデルが異なるために不連続な関節角指令値が必要となる．加えて，ロボットが実現する関節角は破損部で微分不可能となり，線形・非線形関数を組み合わせた微分可能な連続関数でモデル化する適応制御は適用できない．また，先の Peters らの手法と同様に，モデル化のための事前情報の設計やサンプルの与え方に専門家の知識が必要となり，想定できない構造変化への適用は困難と考えられる．今後のサービスロボットへの適用を考慮すると，場合分けによるモデルを切り替えが必要となる構造変化に対しても，自律的に適応可能な手法が望まれる．

非線形かつ微分不可能な点が生じる構造変化に対しては，モデルを考慮せず，構造変化後にタスクを達成する関節角指令値を探索して発見する手法で動作適応が可能と考えられる．しかしながら，探索には膨大な試行回数が必要となるため，実時間での動作適応は困難であった．我々は，動作内での探索すべきポイントを限定して適応する手法⁽¹⁴⁾を提案し，一般的な探索手法に比べて約 10 分の 1 の試行回数で動作を適応した．しかし，既存の探索手法に即しているため，自由度の増加に伴って試行回数が指数関数的に増加する問題が生じ，実時間での動作適応には至っていなかった．そこで更なる試行回数の低減を目指し，本論文では，構造が変化する前のロボットの基本モデルは事前に用意可能である点に着目し，基本モデルと変化後の実ロボットの差異を補償する関節角を逐次学習する手法を提案する．ギアの破損など，数式的には場合分けを必要とする構造変化に対して，構造変化を補償する関節角の差分量を関節角座標上に離散的に学習することで適応を試みる．既知のモデルを有効利用して学習し，不要な試行を削減することにより，一般的な探索手法に比べて高速な手法を提案する．

本論文では，基本的な制御方式の一つである軌道制御によるタスクに限定する．三次元計測機器やセンサデバイスの発達を背景に，カメラや磁気センサ等によってロボットがエンドエフェクタの位置を直接観測できると仮定する．サービスロボットにおいては，動作環境が不定な点や人に対してサービスを行う点から，様々なセンサにより周辺の状況を取得すると考えられる．これらを応用し，自身のエンドエフェクタの位置を観測することとする．ロボットは，実現すべき軌道（目標軌道）とは異なる軌道を観測することで間接的に構造変化を検出する．事前に用意可能な基本モデルに基づき目標軌道を達成する関節角が計算可能な点を活かし，構造が変化した状態で目標軌道を達成する動作を高速に生成する．目標軌道との誤差から生成された動作の精度を評価する．加えて，センサによるエンドエフェクタの位置観測回数の低減が探索時間の短縮につながる点から，動作生成に要する位置観測回数から高速性を評価する．目標軌道との誤差と位置観測回数を既存の探索手法と比較し，提案手法の有効性を確認する．

2. 問 題 設 定

本論文では軌道制御に基づくタスクに着目し，予期せぬ機械的な構造変化がロボットに生じてても，所望の軌道（目標軌道）に可能な限り一致する軌道（適応軌道）を実現する動作を高速に生成することを目的とする．エンドエフェクタの位置座標系において目標軌道 S を実現するため，軌道中に T 個の目標位置を設定する．目標位置 $s(t)$ を $t=1$ から $t=T$ まで順に辿ることで，目標軌道 S を実現する．なお T は目標位置間を移動する際に許容される位置誤差に基づき設計者が事前に与える．ロボットは事前に用意された逆運動学モデル f^{-1} に基づき，目標位置 $s(t)$ を実現する関節角指令値 $\theta(t) = [\theta_1(t), \theta_2(t), \dots, \theta_N(t)]$ （基準関節角）を式 (1) のように計算する⁽¹⁵⁾．なお N はロボットの自由度を表す．

$$\theta(t) = f^{-1}(s(t)) \quad (1)$$

Fig. 1 Position control system on default and changed structure

Fig. 2 Motion adaptation for service robots based on adaptable algorithm

式 (1) は位置座標系から関節角座標系への f^{-1} による写像と考えられ、ロボットの順運動学が f であれば、基準関節角 $\theta(t)$ をロボットに指令することで、ロボットのエンドエフェクタが目標位置 $s(t)$ を実現する (図 1(a))。

ここで構造変化によりロボットの順運動学 f が予期せぬ g へ変化したとする (図 1(b))。事前に用意可能な順運動学モデル f と変化したロボットの順運動学 g は異なるため、基準関節角 $\theta(t)$ を指令しても目標位置 $s(t)$ とは異なる位置 $s'(t)$ が観測される (式 (2))。

$$s'(t) = g(\theta(t)) \quad (2)$$

既存手法では変化後の逆運動学 g^{-1} のモデルを同定することで、構造変化後に目標位置 $s(t)$ を実現する新たな関節角指令値 $\theta_m(t)$ (適応関節角) を式 (3) で計算していた⁽⁹⁾。変化後の逆運動学 g^{-1} が精度よくモデル同定できれば、式 (4) のように、適応関節角 $\theta_m(t)$ をロボットに指令することで、エンドエフェクタが目標位置 $s(t)$ を実現する。

$$\theta_m(t) = g^{-1}(s(t)) \quad (3)$$

$$s(t) = g(\theta_m(t)) \quad (4)$$

予期せぬ構造変化を考慮する必要があるサービスロボットでは、ギア破損のように g を数式で記述する際に破損部を境に場合分けを要する場合がある。我々は、事前にロボットに設計されていた全ての基準関節角に対応する適応関節角が生成できれば、ロボットがタスクを達成できると考える。場合分けやモデル切り替えを必要としない適応手法を提案する。

ここで我々は、ある基準関節角 θ に対する適応関節角 θ_m が一対一で存在する問題に限定する。一方で多自由度ロボットでは逆運動学の解が複数存在するため、目標位置 $s(t)$ に対して基準関節角 $\theta(t)$ や適応関節角 $\theta_m(t)$ が複数存在する可能性がある。我々は、ある目標位置に対して事前に計算する基準関節角は一意であり、その基準関節角に対してある一つの適応関節角が得られることを前提として問題の解決を試みる。なお、関節固着のようにある目標位置に対して複数の適応関節角が考えられるような構造変化 (s, s' に対し θ_m, θ が多値)、そもそも目標位置が可達範囲外になるような構造変化 (s が g の値域外) は、上記前提を満たさないため本論文では対象としない。例えば、二足ロボットにおいて一脚がもげてしまった場合や断線によってトルクが完全に抜け落ちた場合、関節固着が考えられ、ロボットはそもそも姿勢を変化させることができず、目標軌道は実現できない。

3. 提 案 手 法

事前に用意可能なロボットの順運動学モデル f を利用した動作適応システムを図 2 に示す。本論文では、変化

Fig. 3 Learning $\hat{r}(\theta)$ based on observable $r(\theta')$

後の逆運動学 g^{-1} を同定せず，実ロボット g と事前に用意された順運動学モデル f との相違に基づいて，適応関節角 θ_m を得ることを考える．式 (1)，(3) から式 (5) が得られる．

$$\theta_m(t) = g^{-1} \cdot f(\theta(t)) \quad (5)$$

適応関節角 $\theta_m(t)$ は基準関節角 $\theta(t)$ の $g^{-1} \cdot f$ による写像と見なせる．我々は写像 $g^{-1} \cdot f$ を基準関節角 θ に依存する差分ベクトル r と θ との和で表現できると仮定し，式 (6) として定義する．

$$\theta_m(t) = g^{-1} \cdot f(\theta(t)) = \theta(t) + r(\theta) \quad (6)$$

これにより， $\theta(t)$ に対する差分ベクトル $r(\theta)$ を求める問題となる．

差分ベクトル $r(\theta)$ は，関節角座標上でのある点 θ において構造変化を補償するベクトルとなる．微分不可能な点を除くと，関節角座標上の近傍点では差分ベクトルは類似すると考えられる．そこで，関節角座標上の幾つかの点において差分ベクトルを計測し，得られた差分ベクトルから近傍学習法を用いて所望の位置に対する差分ベクトルの推定値（推定差分ベクトル）を計算する．一方で，ロボットに生じる構造変化によっては， θ の十分近傍に位置する差分ベクトルが計測できない可能性があり，その場合，真値とは異なる適応関節角が計算される．そこで，求めた適応関節角を用いて再度動作を試行し，新たに差分ベクトルのデータを取得，近傍学習によって推定差分ベクトルを更新する繰り返し学習法を用いることで，適応関節角を真値へと収束させる手法を提案する．以下，3.1 で近傍学習法を 3.2 で繰り返し学習法を説明する．

3.1 動作の試行と近傍学習

図 3 に計測可能な差分ベクトル r の取得と，基準関節角 θ に対する推定差分ベクトル $\hat{r}(\theta)$ の算出手順を示す．左図は関節角座標を，右図はエンドエフェクタの位置座標を表し，それぞれの座標系における点は運動学により対応付けされる．図 3 中の①のように，基準関節角 θ は，構造変化前にはロボットの順運動学 f により目標位置 s を実現する．構造変化後に基準関節角 θ を指令すると，②のように変化後のロボットの順運動学 g により位置 s' が実現される．この場合，③のように構造変化後に目標位置 s を実現する θ_m^{**} を見つける必要がある．ここでロボットが観測した s' に着目する．既知の逆運動学モデル f^{-1} を用いると，構造変化前に s' を実現する関節角 θ' が計算できる（図 3④）． θ' から θ への差分ベクトル $r(\theta')$ を用いると，構造変化前に関節角 θ' で実現された s' は，構造変化後に $\theta' + r(\theta') (= \theta)$ で実現できると言える． s と s' ，および θ と θ' に十分な近傍関係があれば， $r(\theta) \simeq r(\theta')$ と近似できる．ある一つの基準関節角 $\theta(t)$ を実現すると，位置 $s'(t)$ の観測から $r(\theta'(t))$ がサンプルとして得られる．我々は，関節角座標上で大域的なデータを得ることで $\hat{r}$ の不連続面の把握が早まり，動作全体の修正に有効と考える．一度動作を実行して得られる T 個の大域的な $r(\theta')$ のデータから各々の $\theta(t)$ における $\hat{r}(\theta)$ を近傍学習で求めていく（図 3⑤）．

以下，具体的な近傍学習法を説明する．動作を一度試行して得られる T 個のサンプルから任意の θ に対する $\hat{r}(\theta)$ を近傍学習により計算する． k を学習ステップ， K を総学習回数として，式 (7) で $\hat{r}^*(\theta)$ を更新する．

$$\hat{r}_{k+1}^*(\theta) = \hat{r}_k^*(\theta) + \alpha(k) h_c(t, k) \{r(\theta'(t)) - \hat{r}_k^*(\theta)\} \quad (7)$$

ここで， $\alpha(k)$ は学習係数を表し，式 (8) で定義する．

$$\alpha(k) = \alpha_0 \left(1 - \frac{k}{K}\right) \quad (8)$$

Fig. 4 Convergence based on iteration of $\hat{r}(\theta)$ estimation

$h_c(t, k)$ は近傍関数を表し，ガウス関数に基づき式 (9) で定義する．

$$h_c(t, k) = \exp\left\{-\frac{\|s'(t) - s\|^2}{2\sigma(k)^2}\right\} \quad (9)$$

$h_c(t, k)$ は目標位置 s と観測位置 $s'(t)$ との位置誤差が少ないほど大きな重みを与える．さらに学習ステップ k の増加に伴い標準偏差 $\sigma(k)$ を減少させ，重みの影響範囲を小さくする．単調減少関数 $\sigma(k)$ は初期値 σ_0 と最終値 σ_K を用いて式 (10) で定義する．

$$\sigma(k) = \sigma_0 \left(\frac{\sigma_K}{\sigma_0}\right)^{\frac{k}{K}} \quad (10)$$

$k = K$ で学習終了とし，近傍学習後の差分ベクトル $\hat{r}(\theta)$ を式 (11) で与える．

$$\hat{r}(\theta) = \hat{r}_K^*(\theta) \quad (11)$$

上記の学習手順は自己組織化特徴地図 (SOM) ⁽¹⁶⁾ の学習法に類似する．獲得されるものが 2 次元マップ上の重みではなく多次元の関節角座標上の差分ベクトルという違いはあるが，基本的なアルゴリズムは SOM と同様である．

3.2 繰り返し学習による真値への収束

構造変化によっては，実現すべき目標位置 s に対して十分近傍の s' が観測されず，図 3 のように近傍学習後の $\hat{r}(\theta)$ から求めた θ_m と真値 θ_m^{**} が異なる．そこで，3.1 の近傍学習で計算される θ_m を試行して位置を観測し，観測から得た新たな $r(\theta')$ から $\hat{r}(\theta)$ を再度更新する．動作の試行と近傍学習を繰り返す行うことで， θ_m を真値 θ_m^{**} へと収束させる．図 4 に， $\hat{r}(\theta)$ の繰り返し学習の概要を示す．図 4①のように， i 回目に求めた $\theta_{m,i}$ と真値 θ_m^{**} に誤差が存在すると， $\theta_{m,i}$ を指令値としてロボットが g で実現する位置 $s'_{m,i}$ と目標位置 s にも誤差が生じる．そこで位置 $s'_{m,i}$ を観測し，図 4②のように既知の基本モデル f^{-1} を用いて $\theta'_{m,i}$ を計算する．指令した $\theta_{m,i}$ と観測から計算した $\theta'_{m,i}$ から差分ベクトル $r(\theta'_{m,i})$ を計算する (図 4③)． i 回目の動作の試行結果から得られる T 個の差分ベクトル $r(\theta'_{m,i})$ を用いて，図 4④のように $\hat{r}_{i+1}(\theta)$ を 3.1 の手法で再度近傍学習し， $i+1$ 回目の $\theta_{m,i+1} = \theta + \hat{r}_{i+1}(\theta)$ を生成する． $\|s' - s\|$ が十分小さくなるまで動作の試行と近傍学習を繰り返す．

繰り返し学習の具体的なアルゴリズムを説明する． i 回目に試行した動作から得られる差分ベクトルから， $i+1$ 回目の推定差分ベクトル $\hat{r}_{i+1}$ の算出法を説明する．近傍学習の初期値を式 (12) で与える．

$$\hat{r}_0^*(\theta) = \begin{cases} \{0\}, & i = 0 \\ \hat{r}_i(\theta), & i > 0 \end{cases} \quad (12)$$

前述の近傍学習法で $\hat{r}_K^*(\theta)$ を計算し， $i+1$ 回目の推定差分ベクトル $\hat{r}_{i+1}$ を式 (13) で計算する．

$$\hat{r}_{i+1}(\theta) = \hat{r}_K^*(\theta) \quad (13)$$

動作の試行と推定差分ベクトルの近傍学習を繰り返し行い，真値 θ_m^{**} への収束を試みる．

4. 実 験

提案手法の有効性をシミュレーションにより検証する．家庭用のペットロボットやセラピーロボットとしての応用が期待される四脚歩行ロボットの歩行動作を対象とする．現在，歩行動作はバランスや ZMP に基づき歩行パ

Fig. 5 Target 4-legs robot model on ODE

Table 1 DH parameters of default robot right rear leg

	a	α	d	θ
$XYZ \rightarrow J_1$	0	$-\frac{\pi}{2}$	0	0
	0	0	0	$\frac{\pi}{2}$
$J_1 \rightarrow J_2$	0	$-\frac{\pi}{2}$	0	θ_1
$J_2 \rightarrow J_3$	$-l_2$	$\frac{\pi}{2}$	0	θ_2
$J_3 \rightarrow \text{End-effector}$	$-l_3$	0	0	θ_3

Fig. 6 Target trajectory and observed initial trajectory

ターンを生成する手法が一般的であるが、本論文では生成された脚先軌道パターンにのみ着目する．設計時の脚先の目標軌道を構造変化後にも実現するよう、歩行動作の回復を試みる．

予期せぬ障害物との衝突によるフレーム変形、及び、ギアの破損による構造変化を対象とし、変化した脚の脚先軌道を回復する．シミュレータ ODE⁽¹⁷⁾上にモデル化した四脚歩行ロボットを図5に示す．各脚3自由度を持つ．ロボット脚座標系に対する脚先までの順運動学を DH パラメータ⁽¹⁸⁾で表す．DH パラメータはマニピュレータの構造を表す際によく用いられる表現で、図5の右後脚座標系における脚先位置までの DH パラメータは表1となる．XYZ が右後脚座標系の原点位置、 $J_1 \sim J_3$ が各々関節にあたる．各脚の軌道は各々独立に計算されるため、以降右後脚にのみ着目する．

歩行動作の中でも安定したクロール歩行パターンを設計した．歩行パターン1周期の軌道中に100個の目標位置を設定した ($T = 100$)．設計した右後脚の脚先軌道を図6に示す．図6の横軸が $t (1 \leq t \leq 100)$ 、縦軸は右後脚座標系における脚先位置を示す．脚先位置の X 成分、Y 成分、Z 成分をそれぞれ赤色、青色、緑色で示す．また実線が目標軌道、破線が構造変化前のロボットが実際に実現した軌道を示す．目標軌道を実現するために、ロボットは内部に持つ逆運動学モデル f^{-1} に基づき関節指令値を計算する．実ロボットを想定したシミュレーションであるため、ロボットは地面を歩きながらエンドエフェクタの位置観測を行う．そのため歩行時の外乱やモデル化誤差によって、目標軌道と観測された初期の軌道に僅かながらも誤差が見られた．この誤差は、実際に動作を実行して初めて現れる誤差であり、動作設計段階では想定することが困難な誤差である．厳密なモデル化や動作の調整により誤差を減少させることができるものの、設計段階でこの誤差を減らすことは現実的には難しい．図6を見ると十分に目標軌道と一致しており、かつ、実際に歩行性能には影響しないことが確認できたため、以降、この初期の軌道における誤差を基準として、動作適応時の誤差を比較する．

近傍学習に用いるパラメータは、 $K = 2 \times 10^4$ 、 $\sigma_K = 5 \times 10^{-5}$ 、 $\alpha_0 = 0.8$ とした．なお σ_0 は目標位置 $s(t)$ と観測

Fig. 7 Target 4-legs robot with frame distortion

Table 2 DH parameters of changed right rear leg

	a	α	d	θ
$J_3 \rightarrow \text{End-effector}$	$-l_{3a}$	0	0	θ_3
	0	$-\frac{\pi}{2}$	0	0
	$-l_{3b}$	0	0	$\frac{\pi}{3}$

Fig. 8 Changed trajectory with frame distortion

した位置 $s'(t)$ との位置誤差に基づき式 (14) とした .

$$\sigma_0 = \max_{t \in T} \|s'(t) - s(t)\| \quad (14)$$

4.1 フレーム変形に対する動作適応

構造変化として、右後脚の下腿フレームの変形を仮定し、下腿フレームの中心で X 軸負方向に 60 度変形させた (図 7) . 変形後のロボットの関節 J_3 からエンドエフェクタまでの DH パラメータは表 2 となる . ただしロボットにとって表 2 の情報は未知である .

構造変化後に実現された右後脚の脚先軌道を図 8 に示す . 実線が目標軌道を、破線が観測された軌道を示す . 軸・緑色は図 6 に準ずる . 構造変化により、事前に用意可能な基本のモデルと実ロボットとの対応が取れなくなり目標軌道とは異なる軌道が観測された . 観測された軌道では身体のパランスが崩れ、正しく前進歩行できないことが確認された . 観測された軌道に基づき動作適応を行い、目標軌道の実現を目指す .

脚先位置を観測しながら歩行パターンを実行することで、1 周期で 100 点の位置情報が得られる . 提案手法を用いて 1 周期ごとに修正した歩行パターンを実行して観測した脚先軌道と目標軌道との誤差を図 9(a)(b) に示す . 横軸は観測回数、縦軸が目標軌道との位置誤差 $\|s'(t) - s(t)\|$ の XYZ 各成分を示す . XYZ 各成分ごとの誤差をそれぞれ赤線、青線、緑線で示す . 図 9(a) では初期の誤差の減少を、(b) では誤差の収束を確認するために縦軸のレンジを変えている . 構造変化直後の 1-100 回目の位置観測 ($i = 0$) に基づき適応関節角を求めた . 得られた適応関節角を指令し、動作を実行した結果が 101-200 回目のデータとなる . また同様に、101-200 回目の観測 ($i = 1$) を基に適応した結果が 201-300 回目のデータとなる . このように、直前の 100 回の観測に基づき動作を適応した結果を示している . X 軸成分の誤差を見ると、構造変化直後 1-100 回の観測 ($i = 0$) では正方向の誤差が観測されたが、その情報に基づき修正した 101-200 回 ($i = 1$) では負方向の誤差になった . 観測した 100 点全ての位置が目標位置に対して一方向に偏っていたため、偏りを過度に補償する推定差分ベクトルが推定されたと考えられる . 一方で

Fig. 9 Position errors of generated trajectories with frame distortion

Fig. 10 Position errors of initial and adapted trajectories from target trajectory

誤差の絶対値は減少しており、修正により段階的に位置誤差が0に収束することが確認できる。Y軸成分の位置誤差でも、修正する毎に正負に誤差が変化しているものの、500回の観測後に位置誤差は0収束している。Z軸成分でも同様の結果が得られた。

目標軌道との位置誤差を図10に示す。横軸が t 、縦軸が各 t における目標軌道との位置誤差を示す。細線が構造変化前に基準関節角を指令した場合に観測された初期の軌道における位置誤差を、太線が500回の観測後に生成された適応軌道における位置誤差($i=5$)を示す。前述の通り、基本モデル f のモデル化誤差や歩行に伴う予期せぬ外乱によって、歩行への影響はないものの、構造変化前の初期の軌道においても細線のような誤差が発生していた。設計時には減少させることが困難な細線の誤差を比較基準とし、適応軌道の誤差を議論する。図10より、全ての t において初期の軌道での位置誤差を十分下回る誤差になった。また、最大で1.4mm生じていた位置誤差が0.3mmまで低減され、より目標軌道に近づいた軌道となっている。数式では表現できなかった、モデル化誤差や外乱により元々存在していた誤差も併せて低減させたと考えられる。構造変化前のモデル化誤差も含めた g に対して、十分に適応したと言える。歩行距離を計測して評価を行った。フレームの変形によって構造変化前の34%程度まで歩行距離が減少した。提案手法により歩行距離が約94%まで回復され、歩行性能の面でも十分に適応ができたと考えられる。

脚先位置の観測回数と平均脚先位置誤差を表3に示す。関節角と位置をつなぐ g の情報を得るには、実際に関節角を指令してロボットを駆動させ、エンドエフェクタの位置を観測する必要がある。提案手法では動作の周期毎に学習し、適応関節角を得るため100回の位置観測毎に歩行パターンが更新される。一方で、一般的な探索手法では全ての目標位置に対する探索が終了して初めて歩行パターンが更新される。時間的連続性を考慮した全探索

Table 3 Number of observations and average of position errors

	Ave. of errors [mm]	Number of observations
Default Structure	0.6	-
Changed Structure	149.9	-
All searched	0.02	600000
Conventional	0.5	48000
Proposed($i = 1$)	60.0	100
Proposed($i = 3$)	3.3	300
Proposed($i = 5$)	0.2	500
Proposed($i = 10$)	0.07	1000

Fig. 11 Changes of J_3 joint angles with gear crack

手法では探索終了までに 60 万回の観測回数が必要であった．推定手法を用いて探索する位置の数を減らした手法⁽¹⁴⁾でも約 5 万回の観測回数を必要とした．これらは，実際にロボットを駆動させて位置観測するには膨大な数であった．ロボットが 1 秒間に 3 回関節角の実現と位置の観測が可能であると仮定すると，60 万回の位置観測には 55 時間以上，探索点数を減らした従来法⁽¹⁴⁾では 4 時間半の時間を適応に要する．提案手法では，従来法よりも 2 桁少ない 500 回の位置観測により目標軌道に対する平均位置誤差が 0.2mm となった．初期の平均位置誤差 0.6mm を下回り，目標軌道と十分一致した軌道が得られたと言える．500 回の観測回数で十分に位置誤差を低減することができ，全探索に対して観測回数を 1000 分の 1 に，従来法に対して 100 分の 1 に抑えることが出来た．前述の時間の仮定で考えると，3 分程度で構造変化に適した動作が生成できると見込まれる．

4.2 ギアの破損に対する動作適応

構造変化としてギアの破損を想定し，不連続な関節角指令値を生成する必要がある場合を考える．具体的には右後脚の膝関節 J_3 におけるギア欠けを仮定し，関節が $-100 \sim -95[deg]$ で空転する変化を取り扱う．関節角指令値 θ_3 に対して実際にロボットが実現する関節角（実角度）を ϕ_3 とし，ギア欠けに伴う空転を式 (15) のようにモデル化した．

$$\phi_3 = \begin{cases} \theta_3, & (\theta_3 \leq -100[deg]) \\ -100[deg], & (-100 < \theta_3 \leq -95[deg]) \\ \theta_3 - 5[deg], & (-95[deg] < \theta_3) \end{cases} \quad (15)$$

厳密にギア欠けをモデル化するには，動力学計算も必要となる．しかし本論文の範囲では，2 章の問題設定および 3 章の仮定に基づくため，一様な歩行面において同一の脚先軌道の実現を目指す．そのような問題設定であるため厳密なモデル化は不要と考え，式 (15) の構造変化に対して適応する．ギア欠けに伴う指令値 θ_3 と実角度 ϕ_3 を図 11 に示す．横軸が t ，縦軸が関節角，網掛け部分が空転範囲を示し，実線が各 t においてロボットに指令される基準関節角 θ_3 を，破線が実角度 ϕ_3 を示す．基本の逆運動学モデル f^{-1} で計算される θ_3 は，図 11 の実線のように $-105 \sim -90[deg]$ の角度域で指令されていた．そのため，仮定したギア欠けによる影響を大きく受ける．ロボットは地面に接地しているため，地面からの反力により $-100[deg]$ より大きい場合に空転部が縮退し，破線のような実角度となった．基準関節角がそのまま利用可能な箇所と修正すべき箇所が混在し，不連続な関節角指令値の生成が必要となる．ギア欠け時の指令値と実角度の違いにより，図 12 のように脚先の軌道が変化する．軸，線種，緑色は図 8 に準ずる．空転範囲に当たらない網掛け部分では目標軌道が実現されているが，空転範囲を境に関節角の指令値

Fig. 12 Changed trajectory with gear crack

(a) Position errors on each axis

(b) Details on position errors, enlarged around $\pm 10\text{mm}$

Fig. 13 Position errors of generated trajectories with gear crack

と実角度にずれが生じ、その影響によって軌道もずれた。

提案手法を用いて 1 周期ごとに修正した歩行パターンを行い観測した脚先軌道と目標軌道との位置誤差を図 13(a)(b) に示す。軸、線種、(a)(b) の位置付けは図 9 に準ずる。1-100 回目 ($i=0$)、101-200 回目 ($i=1$)、201-300 回目 ($i=2$) と周期毎に誤差を比較していくと、周期ごとで誤差が減少する部分 (t) が異なるものの段階的に誤差が減少していることが確認できる。前節のフレーム変形では周期毎に各成分の位置誤差が全体的に減少していったが、ギアの破損では空転範囲を境に不連続な差分ベクトルが必要となり、推定差分ベクトルの学習が部分的に進んだと考えられる。繰り返し学習により 300 回の観測回数 ($i=3$) で目標軌道に対する平均位置誤差が 0.2mm となり、十分に小さい位置誤差に抑えることが出来た。

目標軌道との位置誤差を図 14 に示す。軸及び線種は図 10 に準ずる。 t が 20~40 の間に誤差の大きな部分が 2 箇所ある。これらが空転範囲の境界になる箇所であり、境界の前後に存在するサンプルから近傍学習したため度度中間値の適応関節角が生成されていた。 T を増加させて t の間隔を細かくすることで誤差の低減も見込まれるが、構造変化前の初期の軌道に存在した位置誤差と同程度の誤差に収まっており、問題はないと考えられる。他の t でも初期の軌道より誤差を低減することができ、場合分けが必要な構造変化に対しても、最大でモデル化誤差と同程度までの誤差に抑えることができた。歩行距離を計測したところ、ギア欠けによって構造変化前の 75% 程度に減少した歩行距離が、提案手法により 95% まで回復された。構造変化後に目標軌道を回復するだけでも、十分なトルクと T があったため、歩行性能が回復できた。

目標位置 s に対する基準関節角 θ と 500 回の観測から生成された適応関節角 θ_m の θ_3 を図 15 に示す。横軸が t 、縦軸が関節角、網掛け部が空転範囲を示し、細線が各 t における基準関節角の θ_3 を、太線が適応関節角の θ_3 を示

Fig. 14 Position errors of initial and adapted trajectories from target trajectory

Fig. 15 Generated modified joint angle(θ_3)

す．網掛け部に含まれて空転の影響を受けた基準関節角に対し，適応関節角では縮退を考慮して空転の影響を打ち消す θ_3 となった．関節角指令値を不連続に修正する必要がある場合でも提案手法は有効に機能したと考えられる．

5. 考 察

既存の探索手法に比べて少ない探索量での動作適応を実現した．ここでは，既存の探索手法と比較したときの手法の適用範囲を考える．提案手法の性質上，一般的な探索手法と比較して，適用可能な範囲が限定される．これは，既知の逆運動学モデル f^{-1} を利用して構造変化前に s' を実現する関節角 θ' を求める過程に起因しており，逆運動学計算時に解が存在することが前提となるためである．すなわち， s' が f の値域に含まれている必要がある．これが満たされない場合， θ' が計算できないため，差分ベクトルのサンプルが得られない．解が存在する範囲でのサンプルのみから学習することも可能ではあるものの，学習の収束性に対する検討は十分に行えておらず，現時点では収束性が保証できない．同様に，関節固着のように g が一対多になるような構造変化に対しても，一つの目標位置 s に対して様々な差分ベクトルが計算されるため，学習が収束しない． s' が f の値域に含まれ，かつ，一対一の対応となる構造変化であれば，高速な動作適応が期待できる．なお，3章では f が数式に基づく運動学モデルであることを前提に説明していたが， s' に対する θ' が計算可能なモデルがあれば，数式以外のモデルであっても適用できる．加えて，制御すべき変数と関節角を繋ぐモデルがあれば，軌道制御以外に対しても適用できると考えられる．他のタスクへの適用も期待できる一方で， s' が f の値域に含まれるという条件，及び，一対一の構造変化という条件から，既存の制御手法と同様の制限を受けることとなる．実ロボット g とモデル f との差分を更新しながら制御を行う観点で見ると，一種の適応制御の手法として見なすこともできるため，現在適応制御が利用されている範囲であれば適用可能と考える．構造変化の観点から見ると，離散的に差分ベクトルを学習することで，不連続な関節角指令値が必要になる構造変化に対しても，事前に設計された動作を適応することが可能になったと考えられる．

一方，既存のモデル f を用いた逆運動学の計算過程を監視することで，ロボットは容易に s' が f の値域に含まれるかを判断できる．また，誤差の収束性を確認することで，関節固着のような一対多の構造変化が生じているかを判断できる．これらに対しては， f を必要としない既存の探索手法と組み合わせることにより，適用可能な構造変化の範囲を広げることが可能と考えられる．既存手法と提案手法の特徴や適用範囲を活かす組み合わせ方を検討し，今後，ロボットに生じる構造変化に幅広く適用可能な実時間動作適応手法へと発展させていく．

6. む す び

ロボットの構造変化に対し、動作の試行と推定差分ベクトルの更新を繰り返し行い、事前に用意可能な基本モデルと実ロボットとの差異を関節角座標上の差分ベクトルとして逐次学習する手法を提案した。複雑なサービスロボットのモデル同定を行うことなく、事前に用意可能な逆運動学モデル f^{-1} を用いることで、既存の探索手法と比べて高速かつ同程度の精度での動作適応を実現した。適応制御では対象とされていなかった構造自体の変化に着目し、関節角座標上の不連続な構造変化に対しても提案手法が有効なことを示した。

一方で、構造変化により事前モデルの逆運動学計算が出来ない場合や、位置と関節角が一對多で対応する構造変化の場合には、本手法や適応制御に基づく手法では対応が出来ないことが判明した。そのような場合でも、提案手法の実行時に対応できない構造変化をロボットが自律的に判断できる可能性が高い。今後、本手法と試行錯誤に基づく手法を組み合わせたアルゴリズムを開発して適用可能な範囲を拡大していく。

文 献

- (1) 経済産業省, 技術戦略マップ 2010 (2010), 経済産業省.
- (2) 浜田利満, 横山章光, 柴田崇徳, “ロボット・セラピーの展開”, 計測と制御, Vol.42, No.9 (2003), pp.756–762.
- (3) 柴田崇徳, “メンタルコミットロボットにおけるシステム・インテグレーション”, 計測と制御, Vol.44, No.11 (2005), pp.751–756.
- (4) Wada, K., Shibata, T., Musha, T., Kimura, S., “Effects of Robot Therapy for Demented Patients Evaluated by EEG”, *IEEE/RSJ International Conference on Intelligent Robots and Systems* (2005), pp.1552–1557.
- (5) Marti, P., Bacigalupo, M., Giusti, L., Mennecozzi, C., Shibata, T., “Socially Assistive Robotics in the Treatment of Behavioural and Psychological Symptoms of Dementia”, *IEEE/RSJ International Conference on Biomedical Robotics and Biomechatronics* (2006), pp.483–488.
- (6) Wada, K., Shibata, T., “Living With Seal Robots - Its Sociopsychological and Physiological Influences on the Elderly at a Care House”, *IEEE Transactions on Robotics*, Vol. 23, No. 5 (2007), pp.972–980.
- (7) Groom, K.N., Maciejewski, A.A., Balakrishnan, V., “Real-Time Failure-Tolerant Control of Kinematically Redundant Manipulators”, *IEEE Transactions on Robotics and Automation*, Vol. 15, No. 6 (1999), pp.1109–1115.
- (8) 橋本雅文, 中村勇気, 渡辺真志, 大場史憲, “非ホロノミックな移動ロボットの故障診断とフォルトトレラント制御”, 日本機械学会ロボティクス・メカトロニクス講演会'04 講演論文集 (2004), 2P1-H-45.
- (9) 小林亮三, 林広大, 鈴木達也, “Production Rule を用いたシステム表現と故障診断への応用”, 日本機械学会ロボティクス・メカトロニクス講演'05 講演論文集 (2005), 1A1-N-051.
- (10) Peters, J., Schaal, S., “Reinforcement Learning by Reward-weighted Regression for Operational Space Control”, *Proceedings of the 24th International Conference on Machine Learning* (2007), pp.745–750.
- (11) Goethals, I., Pelckmans, K., Suykens, J. A. K., Moor, B.D., “Subspace identification of Hammerstein system using least squares support vector machines”, *IEEE Transactions on Automatic Control*, Vol.50, No.10 (2005), pp.1509–1519.
- (12) Ko, J., Fox, D., “GP-BayesFilters: Bayesian Filtering Using Gaussian Process Prediction and Observation Models”, *Autonomous Robots*, Vol.27, No.1 (2009), pp.75–90.
- (13) Deisenroth, M. P., Huber, M. F., Hanebeck, U. D., “Analytic moment-based Gaussian process filtering”, *In Proceedings of the 26th International Conference on Machine Learning* (2009), pp.225–232.
- (14) 舟洞佑紀, 矢野良和, 道木慎二, 大熊繁, “サービスロボットの機械的な構造変化に対する自律的な動作適応”, 日本機械学会論文集 C 編, Vol.78, No.786 (2012), pp.583–594.
- (15) 日本ロボット学会, 新版 ロボット工学ハンドブック, 新版 (2005), pp.280–287, コロナ社.
- (16) T. コホネン, 自己組織化マップ (1996), シュプリンガー・フェアラーク東京株式会社.
- (17) 出村公成, 簡単! 実践! ロボットシミュレーション Open Dynamics Engine によるロボットプログラミング (2007), 森北出版株式会社.
- (18) 内山勝, 中村仁彦, 岩波講座 ロボット学 2 ロボットモーション (2004), pp.79–130, 岩波書店.