人と接するロボットのための並列タスク設計への最適化によるアプローチ

小林 祐一^{*1*2} 大西 正輝^{*2*3} 細江 繁幸^{*2} 羅 志偉^{*4}

Approach to Parallel Tasks of Human-interacting Robot using Optimization

Yuichi Kobayashi^{*1*2}, Masaki Onishi^{*2*3}, Shigeyuki Hosoe^{*2} and Zhiwei Luo^{*4}

Robots that interact with humans in household environments are required to achieve multiple simultaneous tasks such as carrying objects, collision avoidance and conversation with human, in real time. This paper presents a design framework of multiple human-interacting tasks to meet the requirement by considering stochastic behavior of humans. The proposed designing method first introduces petri-net for parallel multiple tasks. The petri-net formulation is converted to Markov decision processes and processed in optimal control framework. Multiple task arbitration is resolved by optimization with approximated value functions. Two tasks of safety confirmation and conversation tasks are mutually interacting and expressed by petri-net. Tasks that normally tend to be designed by integrating many if-then rules can be dealt with in a systematic manner in the proposed framework, that is, in a state estimation and optimization framework. The proposed arbitration method was verified by simulations and experiments using RI-MAN, which was developed to do interactive tasks with humans.

Key Words: Human Interaction, Behavior Design, Petri-net, Optimal Control

1. はじめに

人と接するロボットの研究は近年、小学校 [1] [2] や博物館 [3] [4] での実証実験などに見られるように活発に行われている [5]。将来的には家庭環境にロボットが進出することも期待されている。人と接する環境では、ロボットには、対象物の運搬、人との対話、人との接触回避などの目標の異なる複数のタスクを同時並列的に行うことが求められる。また、人と接するロボットには、不確実さを含む状況認識を行うことも求められる。環境や観測の不確実さを扱う研究は、一般論としては部分観測マルコフ決定過程による定式化が知られ [6]、移動ロボットの観測・行動計画を対象とした研究も行われている（例えば文献 [7]）。しかし人が任意に動き回ったりロボットに話しかけたりすることに起因する不確かさの検討はまだ十分でなく、これに効率よく対処

するためには人の挙動のモデル化や推定方法などを考察する必要がある。

複数タスクの実時間実行については、タスクスケジューリングの分野で議論がなされている（例えば文献 [8]）。人と接するロボットの複数タスクの場合、タスクスケジューリングの枠組みでの人と接するロボットのタスクの定式化は十分に研究がなされていない。その理由の一つとして、人間の振る舞いによる不確実さが通常のタスクスケジューリングの枠組みで扱うには複雑で定式化しにくいことがあげられる。

一方、柔軟に複雑なタスクを実行することを目指すロボットアーキテクチャの研究としては、反射的・適応的な振る舞いを実現する包摂アーキテクチャ [9]、反射と熟考のハイブリッドアーキテクチャ [10] や階層型アーキテクチャ [11] などが提案されている。また、近年では、petri-net を用いた行動設計のアプローチとして、複数ナビゲーション行動の選択法 [4]、階層的な例外処理を含んだ制御アーキテクチャ [12] やヒューマノイドの運動生成 [13] [14] などの研究が行われている。これらのアーキテクチャ・タスク設計の研究では、対象とする問題がロボット自身の運動・移動に限定されており、人と接することやそれに伴う不確実性、異なる種類のタスク要請などの要素は検討されていない。ビヘービアを作り込んで埋め込むアプローチ（文献 [9]～[11] など）の場合、対象を移動に限った場合は「目標の場所に移動する」という共通の目標のもとでの複数のビヘービアを階層化するなどの設計が可能であるが、本研究で扱うような目標自体が異

原稿受付 2008 年 5 月 14 日

^{*1}東京農工大学

^{*2}理化学研究所 理研-東海ゴム人間共存ロボット連携センター

^{*3}産業技術総合研究所

^{*4}神戸大学

^{*1}Tokyo University of Agriculture and Technology

^{*2}RIKEN-TRI Collaboration Center for Human-Interactive Research

^{*3}National Institute of Advanced Industrial Science and Technology (AIST)

^{*4}Kobe University

■ 本論文は学術性で評価されました。

なる複数タスクを調停する指針は得られない。また, petri-net を用いるアプローチ [4] [12]~[14] は複雑なタスクの表現や処理に適している可能性があるが, petri-net 表現をもとに目的の異なる複数タスクの調停を考察するという拡張は試みられていない。

本論文では, 人と接するロボットのタスクとして, 人との対話や人の存在確率の推定を用いた接触回避のための周辺監視などを扱う。ロボットには, 人がロボットと衝突しないようにロボット周辺の空間に注意を払い, 同時に人との対話が促されるようにできるだけ人を注視する, という二つの相反する要求が存在する。このようなタスクに対し, petri-net [15] によるモデル化を用いた人と接するロボットタスクの定式化を行う。この複数タスクの petri-net 表現に基づき, 人の振る舞いの確率モデル化, 優先度によるタスク間の調停を用いた制御パラメータの最適化によって複数タスク間の調停を実現する枠組みを提案する。提案する枠組みの利点の一つは, 人と接するロボットの振る舞いの設計をシステマティックにできることである。多くの人と接するロボットの行動設計では離散的な状態表現とそのときの行動や状態遷移の条件設定などが記述されるが, そのような設計だけでは遷移条件をどのようにすれば性能を変えられることができるかが明らかになりにくく改良や解析することが困難であった。それに対して本提案法は, 人の挙動を明示的にモデル化することや目標状態への到達時間という指針で複数タスクの調停を行うため, 設計された行動の解析や改良がしやすいという利点がある。

提案設計方法は, 複数タスクの petri-net による記述, マルコフ決定過程 (Markov Decision Process) への変換と最適化という過程からなる。また実際のタスクに上記の最適化を適用するために, 人の存在確率の推定, 人の発話過程のモデル化などを導入する。提案設計法の概略を 2 章に示す。人と接するロボット RI-MAN [17] の注視・対話・運搬タスクへの適用方法を 3 章で示し, 実験およびシミュレーションによる評価結果を 4 章で述べる。最後に 5 章で考察し, 6 章でまとめる。

2. 複数タスクの定式化と最適化

本研究では, (複数) タスクの進行段階に基づいた粗い状態表現を petri-net ならびにマルコフ決定過程を用いて設計し, その空間において複数タスクの調停を行動のパラメータ調節という形で行うというアプローチをとる。

2.1 ロボットシステムと問題設定

ロボット, 人と環境を含む問題設定の概略を Fig. 1 に示す。ロボットシステムは関節・車輪のアクチュエータ, 頭部のスピーカ, 視覚・触覚・聴覚センサとこれらのハードウェアの制御とセンサ情報処理を行うコンピュータからなる。系全体はロボット, 人と環境からなる。

状態変数 ロボット, 人と環境の状態を表す状態変数は連続変数 $x_c \in \mathbb{R}^{N_c}$ と離散変数 $x_d \in \mathbb{Z}^{N_d}$ からなる。

観測変数 ロボットの観測変数は, センサ入力もしくはセ

Fig. 1 Overview of robot system with environment

Fig. 2 Example of petri-net expression for two tasks i and j

ンサ入力に低次の情報処理を行ったものをさす。連続観測変数 $y_c \in \mathbb{R}^{M_c}$ と離散観測変数 $y_d \in \mathbb{Z}^{M_d}$ からなる。

認識過程 認識過程は, 観測変数から状態変数の推定値 (もしくは状態変数を推定する確率分布) $\tilde{x}_c, \tilde{x}_d$ への写像を与える。例えば, 3.3 節で述べる人の存在確率の推定は $\tilde{x}_c$ の例である。

出力変数 ロボットはアクチュエータやスピーカを通して環境に対して行動を出力する。アクチュエータやスピーカへの出力は連続行動出力 $u_c \in \mathbb{R}^{L_c}$ (例えば関節角の目標角速度) および離散行動出力 $u_d \in \mathbb{Z}^{L_d}$ (例えば発話内容に対応する ID) からなる。

2.2 並行タスクの petri-net による記述

ロボットには, 対話や安全確保などの異なるタスクを並行して実行することが求められている。本研究ではこのタスク表現に petri-net を用いる。petri-net は, 前節で述べた状態変数などと異なり, ロボットの内部でのタスクの進行段階を表現するものである。petri-net が客観的な世界を記述するものではない点に注意されたい[†]。

2.2.1 Petri-net

petri-net は, 離散事象を表現する手法群 [18] [19] のうち, 並行して起こる離散的な遷移を表現するための方法の一つである。その利点は, 並行して起こる複数のタスクをコンパクトに表現し, 同期や抑制などによって並行して起こる遷移間の関係を記述できることである。本研究では, 上記の利点からタスク表現に petri-net を用いる。

petri-net は Fig. 2 に示すように place と transition からなる。place 上に存在する円は token と呼ばれ, transition が発火することにより transition により結ばれた別の place に遷移す

[†]また, これに伴い, petri-net から得られるマルコフ決定過程もロボット内部での状況判断に基づいて得られた表現であり, ロボットと人を含んだ系のマルコフ性を保証するものではない点に注意を要する。

る．ある transition が複数の place に入力を持つ場合，そのすべての place に token が存在するときに transition は発火可能となる．また，inhibition place と呼ばれる place に token がある場合，その place に接続された transition は発火できない．これらにより，petri-net では同期および抑制を表現している．

2.2.2 petri-net を用いた複数タスクの記述

ロボットは，状態変数の推定値 $\tilde{x}_c, \tilde{x}_d$ もしくは状態変数の推定確率分布をもとに複数のタスクを実行する．タスク実行の段階は petri-net の place により表され，token によりロボットが現在タスク実行のどの段階にいるかが表される．タスクの数を n とする．各タスクについて望ましい状態を表現するため，各タスクについて目標 place を定義する．人との衝突を回避するための周辺監視タスクを例にとると，このタスクには，周辺に人がいるのかいないのか不明な段階や近傍に人がいないことが確認できている段階などがあり，後者の段階に達することがこのタスクの目標であるといえる．タスク i の目標 place を d_0^i と表す． m_i をタスク i の（目標 place を含めた）place の数とし，タスク i の place を $d_0^i, d_1^i, \dots, d_{m_i}^i$ と表す．Fig. 2 では，目標 place は二重線で描かれている．一般の petri-net には token の総数に制約はないが，本研究では各タスクについてつねに一つであるように petri-net を設計する．本研究で用いる petri-net は binary petri-net である [20]．すなわち，すべての transition の重み（transition が発火するために必要な token の数）は 1 である．これらの仮定より，token の存在する place はタスクの現在の段階を表す．

本研究で扱うタスクでは，人の挙動に起因する不確実さにより，同じロボットの振る舞いに対して異なる遷移が起きる可能性がある．これを表現するため，単一の place が複数の transition を出力に持つ（競合する）場合も考える．このような場合は，複数の transition のうちどれが発火するかは（発火）確率により特徴づけられるものとする[†]．さらに本研究では，以下の仮定を設ける．

- (1) 各 transition に発火のコストとして期待所要時間を設定する．
 - (2) 各 place に自身へ遷移する transition を設定する．
- transition の発火に遅延時間を設定するモデルとしては時間 petri-net が知られているが，本論文では後のマルコフ決定過程への変換の簡便さのため時間 petri-net ではなく通常の petri-net を用いる．したがって，各 transition は発火条件が満たされれば常に発火可能である．その代わり，(1) に示したように各 transition には遷移時間（発火のコスト）という属性を付与

し，タスクの実行所要時間を評価できるようにしている．(2) は，マルコフ決定過程に変換する際，自分自身への状態遷移を定義することで表現する．一般の petri-net では transition は瞬間的に発火し自身への遷移は定義されないのが普通であるが，本研究ではマルコフ決定過程に基づいた定式化を導入するために以上の仮定を導入した．なお，3 章で述べる petri-net の設計例では，自身への遷移をすべて記述すると煩雑になるため省略する．

2.3 マルコフ決定過程での最適制御

タスク i について，現在 token が存在する place を $s^{(i)}$ と表す．これは各タスクについて，タスク実行の段階を表すものになっている．これを，(2.1 節で述べた状態変数と混同しない限り）タスク状態と呼ぶことにする．全タスクについて総合した状態を $s = \{s^{(i)} | i = 1, \dots, n\}$ と表す．ロボットの行動 a は，例えば首振りや発話など，各タスクを目標タスク状態に向かわせるためにロボットがとる振る舞いを表す．本研究では，タスクごとに異なる行動を定義する^{††}という方法をとらず，全タスクに共通したものとして行動を定義する．行動を状態の関数として $a = \pi(s)$ と表すとき， π を方策と呼ぶ．ここで，transition はタスク状態の遷移に対応するのに対して行動はロボットの物理的な振る舞いを表すものであり，行動と transition が一対一に対応するわけではない点に注意されたい．

本研究で定める petri-net はタスクごとに一つの token のみが動くという仮定を設けているため，その意味では token の遷移確率をタスクごとに定義することが可能である．しかし，同期や抑制などで他のタスクの token の位置によって遷移が制限される場合があるため，原理的には全タスクの token 位置を考慮しないと正確な遷移確率は定められない．本研究では，全タスクの token を考慮して計算量が増加するのを防ぐため，まず同期や抑制を無視した場合のタスクごとの遷移確率を用いて価値関数の計算をオフラインで行い，オンラインでの行動選択時に全タスクの token を考慮した計算を行うという簡略化のアプローチをとる．異なるタスク間での同期と抑制の制約を無視した場合，タスク i についての place 間の token の遷移は，遷移確率と遷移所要時間期待値で特徴づけられる． $p(s_k^{(i)}, s_l^{(i)}, a)$ は，行動 a をとったときの $s_k^{(i)}$ から $s_l^{(i)}$ への遷移確率を表し^{†††}， $r(s_k^{(i)}, s_l^{(i)}, a)$ はその遷移にかかる時間の期待値を表す．この遷移所要時間期待値は，行動により決まるものとする．つまり，

$$r(s_k^{(1)}, s_l^{(1)}, a) = \dots = r(s_k^{(n)}, s_l^{(n)}, a) = T(a) \quad (1)$$

のように行動により決まる時間期待値 $T(a)$ がすべてのタスクについてのタスク状態の遷移に適用されるものとする．**Fig. 3**

Fig. 3 Transformation of petri-net to Markov decision process

[†]一般に「stochastic petri-net (確率 petri-net)」として知られる petri-net は，ある transition の発火の遅延時間が確率分布（指数分布）で与えられるものを表す [15] [16]．本研究で扱う確率は競合する複数の transition のうちどれが発火するかを確率で表現するものであり，一般の確率 petri-net とは異なる．また，本研究では発火の遅延時間の確率分布を特定しなくても遷移時間期待値を特定すれば最適制御問題が計算可能なため，遅延時間の確率分布は特定しない．

^{††}あるタスクのための腕の制御，別のタスクのための頭の制御など，タスクごとに異なる身体部位の制御を同時進行的に行動として定義する方法も考えられる．

^{†††}この遷移確率は，同期・抑制を考慮せずタスクごと独立に定義されたものとする．

のように、同一 place に token がとどまる場合を $s_k^{(i)}$ から $s_k^{(i)}$ への遷移という形で表現するものとする。これにより、前節で述べた仮定 (1) (2) をマルコフ決定過程上で表現できる。

タスク i についての目的は、token をできるだけ短時間で目標 place に到達させる（もしくは目標 place に存在し続けさせる）ことである[†]。離散状態の最適制御の枠組み [22] では、この問題は期待値 $E_\pi \left\{ \sum_{t=0}^{\infty} r_t^{(i)} \right\}$ を最小にするような方策 π を発見する問題として定式化される。ここで、 $r_t^{(i)}$ は時間ステップ t において行動にかかる時間である（ただし、タスク i について目標 place に到達している状況では $r_t^{(i)} = 0$ となるとする）。このような定式化のもとで離散状態の最適制御を扱う枠組みとして強化学習 [21] が知られているが、一般的な強化学習問題の枠組みで導入される割引率 γ はここでは省かれる（ $\gamma = 1$ とする）。

この最適制御問題において、タスク i についての状態価値関数は

$$V_\pi^{(i)}(s_k^{(i)}) = E_\pi \left\{ \sum_{t=0}^{\infty} r(s_k^{(i)}, s_l^{(i)}, a) \right\} \quad (2)$$

で与えられる。ただし、

$$r(s_0^{(i)}, s_0^{(i)}, a) = 0, V_\pi^{(i)}(s_0^{(i)}) = 0, s_0^{(i)} = d_0^i \quad (3)$$

とする。最適価値関数 $V_\pi^{(i)}(s)$ は次式の Bellman 方程式を満足する [21]。

$$V_\pi^{(i)}(s_k^{(i)}) = \min_{a \in \mathcal{A}(s)} \sum_{s_l^{(i)}} p(s_k^{(i)}, s_l^{(i)}, a) \left[r(s_k^{(i)}, s_l^{(i)}, a) + V_\pi^{(i)}(s_l^{(i)}) \right], s = \{\dots, s_k^{(i)}, \dots\} \quad (4)$$

ここで $\mathcal{A}(s)$ は s に対する行動集合である（本来行動 a を決定するには他のタスクについてのタスク状態を考慮する必要があるが、これについては次節で述べる）。遷移確率と所要時間が既知であれば、価値関数 $V_\pi^{(i)}(s^{(i)})$ は上式を用いて計算可能であり、これを用いて最適行動を導出することも可能である。

2.4 行動のパラメータ化と最適化による複数タスクの調停

行動 a はロボットで実行するにあたり、より具体的な表現をとる。すなわち、連続出力 u_c もしくは離散出力 u_d をある一定時間の時系列で出力するものとして表される。例えば首振り行動を具体的に表現する場合、「頭部関節を α から β [rad] まで動かす」という形で行動を $[\alpha, \beta]$ によりパラメータ化することが可能である。このような行動のパラメータ化を一般的に $a(\theta)$ のように表す（この例では、 $\theta = [\alpha, \beta]^T$ となる）。本研究で扱う最適制御問題では、行動 a ではなくパラメータ θ の決定を行う。パラメータ集合は行動に依存し、 $\Theta(a)$ と表す。

行動のパラメータ調節に関する Bellman 方程式は

$$V_\pi^{(i)}(s_k^{(i)}) = \min_{\substack{a \in \mathcal{A}(s), \\ \theta \in \Theta(a)}} \sum_{s_l^{(i)}} p(s_k^{(i)}, s_l^{(i)}, a(\theta)) \left[r(s_k^{(i)}, s_l^{(i)}, a(\theta)) + V_\pi^{(i)}(s_l^{(i)}) \right], s = \{\dots, s_k^{(i)}, \dots\} \quad (5)$$

と表せる。 $a = \pi(s)$ であり、 a は θ の関数であるため、 π もまた θ によりパラメータ化されると考えることができる。これを $\pi_s(\theta)$ 、もしくはより簡単に $\pi(\theta)$ と表す。本研究では、方策 π を固定したまま、パラメータ θ を変更して最適制御の探索を行う。これは周辺監視の首振り行動の例では、「頭部関節を動かす」という基本的な行動は変わらないが、軌道を調節するパラメータ α, β を変更することで首振り行動を調節することに対応している。 $\pi(\theta)$ のもとでのタスク i の状態価値は以下を満たす。

$$V_{\pi(\theta)}^{(i)}(s_k^{(i)}) = \sum_{s_l^{(i)}} p(s_k^{(i)}, s_l^{(i)}, a(\theta)) \left[r(s_k^{(i)}, s_l^{(i)}, a(\theta)) + V_{\pi(\theta)}^{(i)}(s_l^{(i)}) \right], s = \{\dots, s_k^{(i)}, \dots\} \quad (6)$$

パラメータ θ についての最適制御の解を得るために、式 (5) の繰り返し計算をすべての状態や行動調節パラメータについて行うのは、計算量の観点から現実的ではない。そこで、ノミナルなパラメータ $\bar{\theta}$ を導入する。方策 $\pi(\bar{\theta})$ のもとでの価値関数はオフラインで $\theta = \bar{\theta}$ として式 (6) を用いることで計算が可能である。こうして得られた $V_{\pi(\bar{\theta})}^{(i)}(s)$ を用いて、複数タスクの優先度を考慮した最適パラメータの近似的な探索は以下で行うことができる。

$$[\theta|s] = \arg \min_{\theta} \sum_{i=1}^n w_i Q_{\pi(\bar{\theta})}^{(i)}(s_k^{(i)}, a(\theta)) \quad (7)$$

ここで、 $Q_{\pi(\bar{\theta})}^{(i)}(s_k^{(i)}, a(\theta))$ は次式で与えられる。

$$Q_{\pi(\bar{\theta})}^{(i)}(s_k^{(i)}, a(\theta)) \equiv \sum_{s_l^{(i)}} p_s(s_k^{(i)}, s_l^{(i)}, a(\theta)) \left[r(s_k^{(i)}, s_l^{(i)}, a(\theta)) + V_{\pi(\bar{\theta})}^{(i)}(s_l^{(i)}) \right] \quad (8)$$

ただし、 $w_i > 0$ はタスク間の優先度を設定するための重み係数である。また、 $p_s(s_k^{(i)}, s_l^{(i)}, a(\theta)), r_s(s_k^{(i)}, s_l^{(i)}, a(\theta))$ は、全タスクの token の位置 s に依存して決まる遷移確率および遷移所要時間期待値である。この計算では複数タスク間の同期や抑制を考慮するものとする。

ここで、 $Q_{\pi(\bar{\theta})}^{(i)}(s_k^{(i)}, a(\theta))$ の値はタスクごとに計算され、タスク同士で同時に最小化できない可能性のあるものであるのに対し、行動 $a(\theta)$ は全タスクに共通である点に注意されたい。周辺監視タスクと人との対話タスクの例では、求められる首の振り方は異なるが、いずれのタスクに対しても首を振るという行動自体は共通しているということである。式 (7) は、異なるタスクを「目標タスク状態への到達時間」という指標を用いて同じ土俵に乗せることを提案するものである。これにより、最短

[†]petri-net の問題では、最短時間可達問題に相当する。ただし、一般の petri-net とは、transition の発火を制御するのではなく遷移確率や所要時間を a を通じて制御する点が異なる。

時間制御問題の（近似的な）最適性と複数タスク間の優先順位を考慮した行動パラメータの選択が可能になる．実際の適用に際しては，式（8）において p や r が観測値や状態推定値などに依存して変化する．これにより，現在（およびそれまで）の観測に依存した行動パラメータの選択が行われる．

3. 人と接するタスクの RI-MAN への実装

3.1 RI-MAN のハードウェアと変数

RI-MAN は人と接する環境で対象物（人など）を抱えあげるようなタスクを実現することを目標に開発された [17] ロボットである．Fig. 4 に RI-MAN の外観を示す．各腕に 6 自由度，腰に 2 自由度，頭部に 3 自由度[†]，両車輪に 2 自由度を有する．顔前面にスピーカー，耳の位置にマイク，目の位置に CCD カメラ，さらに両上腕・前腕と胸に合計五つの触覚センサを搭載している．

状態変数は以下である．

- 対象物位置（ロボット座標系）： x_o
- ロボットの関節角度： x_{angle}
- 人により与えられる指示： x_{com}
- 人の数： x_{nh}
- ロボット座標系における人の位置： x_{human}

2.1 節の定義との対応をとると， $x_c = [x_o^T, x_{angle}^T, x_{human}^T]^T$ ， $x_d = [x_{com}, x_{nh}]^T$ である．人からの指示はロボットのデータベースに登録されている．ロボットは人の発話を認識したとき，データベースを参照し，登録されている内容であればそれに応じた発話を行う．

観測変数は以下である．

- 画像処理により得られる対象物の（ロボット座標系での）位置： y_o
- 画像処理により得られる人の（ロボット座標系での）位置： y_{human}
- 音源定位処理により得られる，音を出している人の方向： y_{sound}
- 音声認識処理により認識される指示・会話の ID： y_{com}
- 触覚センサにより得られるロボットの身体と人の接触情報： y_{tac}

ここで， $y_c = [y_o^T, y_{human}^T, y_{sound}^T, y_{tac}^T]^T$ ， $y_d = y_{com}$ である．画像処理では，肌色の検出を行い，人の顔を抽出しステレオ視によりその位置を算出する．音源定位処理は両耳部にあるマイクにより行い，音の発生した方向を同定する [25]．触覚センサ情報は，人との接触を検知するのに用いる^{††}．

ロボットの出力変数は以下である．

- 関節角の望ましい角速度： u_{angle}
- 車輪の目標回転速度： u_{wheel}
- 発話内容の ID： u_{speech}

[†]このうち，本実験ではパンとチルトの 2 自由度のみを用いる．

^{††}ただし，人以外の障害物はない，もしくはあってもロボットが完全に位置を把握していると仮定する．

^{†††}ここでは安全確保タスクと対話タスクの並列実行に主に注目するため，ロボットの移動を考慮しない形での情報表現を行っているが，車輪の回転速度情報を用いてロボットの移動を考慮した情報表現にすることは可能である．

Fig. 4 RI-MAN with an object

Fig. 5 Expression of tasks by petri-net

ここで， $u_c = [u_{angle}^T, u_{wheel}^T]^T$ ， $u_d = u_{speech}$ である．

3.2 並列タスクの定義

Fig. 5 に本研究で実装されるタスクを示す．安全確保タスク，対象物運搬タスクと対話タスクの三つが同時に実行される．ただし，対象物運搬のための移動は，対話タスクでの対話結果により開始される^{†††}．なお，place の設定は，まずロボットに求められる振る舞いを設計し，その振る舞いに place を対応させるという指針で行う．

安全確保タスク

ロボットは周囲を見回し，自身の周囲に人がいるかどうかを確認する．近くに人がいると判断したときは，人がそれ以上ロボットに近寄らないよう，話しかけることで注意を喚起する．このために，ロボットは人の存在確率を $p_h(i_1, i_2)$ という形で推定する．ただし $[i_1, i_2]$ はロボット周辺の二次元空間（ロボット座標系）を格子状に分割した格子を表す．人の存在を判定するための閾値として， p_{th1} と p_{th2} ($0 < p_{th1} < p_{th2} < 1$) を導入し，

$$\text{humanexistsat } [i_1, i_2] \quad \text{when } p_h(i_1, i_2) > p_{th2}$$

$$\text{humandoesnotexistat } [i_1, i_2] \quad \text{when } p_h(i_1, i_2) < p_{th1}$$

のように人の存在の判断に用いる．

ロボットの近傍の格子の集合を $R_{vicinity}$ ，ロボットと接触す

Fig. 6 Estimated probability distribution of human existence

る格子の集合を R_{contact} とする。Fig. 6 にロボット周辺空間の格子の集合を示す。格子は R_{vicinity} , R_{contact} およびいずれにも属さない $R_{\text{rest}} = R_{\text{all}} \setminus (R_{\text{vicinity}} \cup R_{\text{contact}})$ の三つに分類される。これらの集合を用いて、安全確保タスクの place を以下のように定義する。

- COLLISION: $p_h > p_{th1}$ となる確率 p_h をもつ格子が $[i_1, i_2] \in R_{\text{contact}}$ に存在する。
- ATTENTION: token は ‘COLLISION’ に存在せず、かつ $p_h(i_1, i_2) > p_{th2}$ となる確率 p_h をもつ格子が $[i_1, i_2] \in R_{\text{vicinity}}$ に存在する。
- UNKNOWN: token は ‘COLLISION’ および ‘ATTENTION’ に存在せず、かつ $p_{th1} < p_h < p_{th2}$ となる確率 p_h をもつ格子が $[i_1, i_2] \in R_{\text{rest}} \cup R_{\text{vicinity}}$ に存在する。
- SAFETY confirmed: token は他の place に存在しない。すなわち、 R_{vicinity} のすべての格子の存在確率は $p_h < p_{th1}$ となる。

place 間の token の遷移は、 p_h が変化することにより、上記の条件に従って token の存在する place が切り替わることで表現される。SAFETY confirmed place が安全確保タスクの目標 place である。COLLISION place はロボットと人との接触が触覚センサにより検知された状態を表している。

対象物運搬タスク

対象物運搬タスクは、対話タスクにおける命令の認識によって移動と運搬を開始する。ロボットは対象物に向かって移動し、対象物正面で停止して対象物を持ち上げる [17]。安全確保タスクの token が COLLISION place にあるとき、ロボットに接触している人を傷つけないようにその状態で停止する。対象物運搬タスクの place を以下のように定義する。

- WAITING for command: 指示者から命令を受けていないとき、人との対話を待つ。
- APPROACH: 命令を受けたあと、ロボットは対象物に接近する。対象物に対して適切な位置・姿勢に到達したとき、token は HOLD UP place に遷移する。
- HOLD UP: 抱えあげ動作が終了したとき、token は COMPLETE place に遷移する。

[†] ロボットの返答が終了すると強制的に FOUND place に遷移するため、定常的に目標 place に滞在するという形ではなく、FOUND place に滞在しながらできるだけ頻繁に SPEAK place への遷移を目指すという形で最適制御の枠組みに適合させる。

- COMPLETE: ロボットは停止し、token を WAITING place に遷移させて指示者からの新たな命令を待つ。

COMPLETE place が対象物運搬タスクの目標 place である。

対話タスク

ロボットは人の発声を認識し、それに対する返答を行う。人の発声がない場合は、ロボットは人の方を向き続けることで人の発話を促す。対話タスクにおいては以下の三つの place が定義される。

- Human FOUND: $p_h(i_1, i_2) > p_{th2}$ となる格子 $[i_1, i_2]$ が存在する。マイクにより発話が認識された場合、この place の token は SPEAK に遷移する。
- SPEAK: 人間の発声の音声認識結果に基づき、ロボットはスピーカーから返答を行う。返答の発話の終了時に token は SPEAK から FOUND へ遷移する。
- Human LOST: $p_h(i_1, i_2) > p_{th2}$ となる格子が存在しない。SPEAK place が対話タスクの目標 place である[†]。FOUND place から SPEAK place への遷移は人の発話に依存する。この過程は、人の確率的な振る舞いのモデル化により表現する。人の発話確率は、ロボットの顔が人の方向を向いている時間に依存して決まると仮定する。これについては 3.5 節で述べる。

3.3 人の存在確率の推定

ロボットは、画像情報および音源定位情報を用いて人の存在確率の推定を行う。この推定の目的は、1) 対話タスクに関連して、人が自然にロボットに話しかけられるように人の方向を向く行動に利用すること、および 2) 安全確保タスクに関連して、ロボットの近辺に人が存在しないように安全の確保を行うこと、の 2 点である。人が存在する位置の推定だけでなく、(ある場所に) 人が存在しないことを表現することが求められる。このため、人数、人の位置などの状態量を推定する代わりに、各位置における人の存在確率の推定を occupancy grid [23] [24] を用いて行う。

存在確率の推定はロボット周辺の (ロボット座標系での) 平面を Fig. 6 のように格子状に分割して行われる。 $p_h(i_1, i_2)$ を格子 $[i_1, i_2]$ における人の存在確率の推定値とする。 p_h の推定は以下の二つの処理によって更新する (各処理では独立に p_h の更新を行う)。

3.3.1 時間発展

人はランダムに移動するという仮定に基づき、以下のように存在確率を更新する。

$$p_{h,t+1}(i_1, i_2) = \sum_{(j_1, j_2) \in R_{\text{all}}} P_{(j_1, j_2)}^{(i_1, i_2)} p_{h,t}(j_1, j_2) + \sum_{(j_1, j_2) \in R_{\text{ext}}} P_{(j_1, j_2)}^{(i_1, i_2)} p_{h,t}(j_1, j_2) \quad (9)$$

ここで、 $P_{(j_1, j_2)}^{(i_1, i_2)}$ は格子 $[j_1, j_2]$ にいる人間が次の時間ステップに $[i_1, i_2]$ に移動する確率を表す。この確率は次式を満たす。

$$\sum_{i_1, i_2} P_{(j_1, j_2)}^{(i_1, i_2)} = 1, \quad \forall (j_1, j_2) \quad (10)$$

R_{ext} は R_{all} より外の仮想的な格子を表す。仮想格子での存在確率は一定の確率 $p_h(i_1, i_2) = p_{\text{ext}}, (i_1, i_2) \in R_{\text{ext}}$ であるとする。

観測更新 画像処理と音源定位処理の結果を用いて存在確率の更新を行う。 ρ_{im} と ϕ_{im} を画像処理により得られた人間までの（ロボット座標系での）距離および角度とする。ノイズがガウス分布に従うという仮定の下、観測モデルは以下のように表される。

$$p(\rho_{im}, \phi_{im} | \rho, \phi) = \frac{1}{2\pi\sigma_r\sigma_{im}} \exp \left[-\frac{1}{2} \left(\frac{(\rho - \rho_{im})^2}{\sigma_r^2} + \frac{(\phi - \phi_{im})^2}{\sigma_{im}^2} \right) \right] \quad (11)$$

ここで $p(\rho_{im}, \phi_{im} | \rho, 0)$ は人間が距離 ρ , 角度 ϕ [rad] に存在するときに観測 ρ_{im}, ϕ_{im} を得る確率を表す。この観測モデルから、各格子 $p_h(i_1, i_2)$ の存在確率が更新される [23]。音源定位処理により得られた人間の方向を ϕ_s とする。音源定位処理における観測モデルは

$$p(\phi_s | \phi) = \frac{1}{\sqrt{2\pi}\sigma_s} \exp \left(-\frac{(\phi_s - \phi)^2}{2\sigma_s^2} \right) \quad (12)$$

と表される。ただし、 $p(\phi_s | \phi)$ は人間が ϕ [rad] の方向で音を発したときに ϕ_s という観測を得る確率を表す。

3.4 最適化による頭部軌道生成

首振り行動は、安全確保タスクのすべての place、および対話タスクの FOUND, LOST place について行われる。この節では、2.4 節で述べた最適化の枠組みで首振り行動の軌道を決定する方法について述べる。

頭部の軌道生成は、 n_{traj} 個の（頭部の pan, tilt 角で構成される空間における）軌道候補を生成したあとで、最適なものを選択するという方法をとる。候補軌道 θ_k ($k = 1, \dots, n_{traj}$) は以下の手順で生成される。Fig. 7 に示すように、各格子の確率 $p_h(i_1, i_2)$ に基づいて、格子を軌道の経由点として確率的に選択する。ある格子が選択されると、その格子に対応する頭部の関節角度（pan, tilt）が計算される。こうして得られる関節角度を経由点として、軌道を構成する。

各格子を経由点として選択する確率は、確率関数 $p_{select}(p_h)$ に基づいて決められる。存在確率の高い格子の選択確率を高くするように設定する。Fig. 8 に選択確率 $p_{select}(p_h)$ の例を示す。 p_h が 1 に近い場合に選択確率は 1 に近くなり、 p_h が 0 に近い場合に 0 に近くなるように設計されている。ロボット頭部の pan と tilt 角 [rad] を $[q_{pan}, q_{tilt}]^T$ とし、 i 番目の経由点の pan-tilt 角を q_i とする。すなわち、 $q_i = [q_{pani}, q_{tilti}]^T$ とする。

Fig. 7 Generation of trajectory with via points

これらの経由点をつなげることにより、軌道は以下で表される。

$$\theta_k = \{q_1, q_2, \dots, q_{n_k}\} \quad (13)$$

ここで n_k は k 番目の軌道候補の経由点数である。ただし、各関節角度は次の関係が成り立つように並べられる。

$$\arctan(q_{tilt1}/q_{pan1}) < \arctan(q_{tilt2}/q_{pan2}) < \dots < \arctan(q_{tiltn_k}/q_{pann_k}) \quad (14)$$

頭部関節の角速度 v_{head} が一定になるように速度を設定し、周期運動の周期は

$$T(a(\theta_k)) = \sum_{j=1}^{n_k} \frac{\|q_{j+1} - q_j\|}{v_{head}}, \quad q_{n_k+1} \equiv q_1 \quad (15)$$

となる。この周期は 2.4 節で述べた所要時間に対応し、

$$r(s_k^{(i)}, s_l^{(i)}, a(\theta_k)) = T(a(\theta_k)) \quad (16)$$

となる。例えば、 $s_k^{(i)}$ は安全確保タスクのすべての place と対話タスクの FOUND, LOST place に対応する。

安全確保タスク（タスクの番号を 1 とする）の場合、遷移確率 $p(s_k^{(1)}, s_l^{(1)}, a(\theta_k))$ は次のように計算される。まず、頭部軌道 θ_k により生成される頭部の運動系列によって視野に入る格子の集合を $R_{visible}$ とする。また、人の存在が未知である（存在確率が閾値 p_{th1}, p_{th2} の間である）格子の集合を $R_{unknown}$ とする。すなわち、

$$R_{unknown} = \{[i_1, i_2] | p_{th1} < p_h(i_1, i_2) < p_{th2}, [i_1, i_2] \in R_{rest} \cup R_{vicinity}\} \quad (17)$$

である。もし θ_k により実現される視野が未知の格子すべてを含まない場合、token は UNKNOWN place にとどまる。これは、以下のように表される。

$$\text{If } R_{visible} \cap R_{unknown} = \emptyset, \text{ then } p(s_2^{(1)}, s_2^{(1)}, a(\theta_k)) = 1, \\ p(s_2^{(1)}, s_1^{(1)}, a(\theta_k)) = p(s_2^{(1)}, s_0^{(1)}, a(\theta_k)) = 0 \quad (18)$$

ただし $s_0^{(1)}, s_1^{(1)}$ および $s_2^{(1)}$ は SAFETY, ATTENTION, UNKNOWN の place をそれぞれ表す[†]。一方、 $R_{visible} \cap R_{unknown} \neq \emptyset$ である場合、遷移確率は次式で与えられる。

$$p(s_2^{(1)}, s_2^{(1)}, a(\theta_k)) = 0 \quad (19)$$

$$p(s_2^{(1)}, s_1^{(1)}, a(\theta_k)) = \prod_{[i_1, i_2] \in R_{visible} \cap R_{vicinity}} (1 - p_h(i_1, i_2)) \quad (20)$$

$$p(s_2^{(1)}, s_0^{(1)}, a(\theta_k)) = 1 - p(s_2^{(1)}, s_1^{(1)}, a(\theta_k)) \quad (21)$$

対話タスクでの遷移確率については 3.5 節で述べる。

3.5 人との対話のモデル化と状態遷移確率

人間の発話は確率的に起こり、発話の確率はロボットが人間をどの程度見ているかに依存すると仮定する。本節ではこの仮定に基づき、人の発話の確率モデルを用いて状態遷移確率を見積もる方法を示す。

[†] 人とロボットの衝突はロボットの頭部軌道を変えても制御することができないため、COLLISION place についての確率は省略している。

Fig. 8 Function of grid selection probability p_{select}

Fig. 9 Function of speech probability p_{speech}

ロボット頭部の動きは周期的であるとし、その周期を 3.4 節で述べたように $T(a(\theta_k))$ とする。ロボット頭部が人間の方向を向いている時間の比率を

$$\xi(\theta_k) = \frac{\tau_{(i_1, i_2)}(\theta_k)}{T(a(\theta_k))}, \quad \text{ahumanexistsingrid}[i_1, i_2] \quad (22)$$

と定義する。ただし、 $\tau_{(i_1, i_2)}(\theta_k)$ は運動周期内に格子 $[i_1, i_2]$ がカメラ視野に含まれる時間とする。人の発話確率は比率 $\xi(\theta_k)$ の関数として $p_{\text{speech}}(\xi(\theta_k))$ のように表せる。**Fig. 9** は $\xi(\theta_k)$ の関数としての発話確率の例である。比率 ξ が 1 に近いときには確率は 1 に近づき、 ξ が 0 に近いときは減少するように定義されている。

対話タスク（タスクの番号を 2 とする）における token の遷移確率はこの発話確率に等しく、

$$p(s_1^{(2)}, s_0^{(2)}, a(\theta_k)) = p_{\text{speech}}(\xi(\theta_k)) \quad (23)$$

である。遷移の期待所要時間は頭部の運動の 1 周期に等しく、

$$r(s_1^{(2)}, s_0^{(2)}, a(\theta_k)) = T(a(\theta_k)) \quad (24)$$

である。

4. 評価実験

提案する最適化の枠組みを RI-MAN において実装し、シミュレーションおよび実験により評価を行った。なお、本問題設定では対象物運搬タスクに遷移があっても、ロボットの移動中に対象物運搬タスクと安全確保タスク・対話タスクの間に競合が生じないため、三つのタスク間の調停の効果を示すことは困難である。提案手法による調停の効果を理解しやすくするため、以下では対象物運搬タスクに関しては token が WAITING place にとどまる場合について限定し、安全確保タスクと対話タスクの間の調停に関する評価を行った。式 (7) における（頭部の）候補軌道数は 10 とした。ノミナルパラメータによる状態価値は、4.2 節で述べるシミュレーションにおいて得られた状態遷移確率をもとに計算した。また、タスク調停の重み係数は $w_1 = w_3 = 1$ とした。

4.1 デモンストレーション実験

RI-MAN のディスプレイ上でのモニタ画像を **Fig. 10** に示す。図中左上部は現在の推定確率分布 $p_h(i_1, i_2)$ を表している。高い確率は濃い色で表すように設定している。白い円はロボット位置を示している。格子のサイズは 0.3 [m] の正方形である。 R_{all} に対応する格子数は 16×16 である。図中には、人の顔が

Fig. 10 Monitoring image of RI-MAN with probability density and camera recognition results

Fig. 11 Safety task with one human moving around the robot

ロボット前方に認識されている。顔の位置に相当する格子近辺では人の存在確率が高くなり、その周辺では低くなっている。視野外の他の格子の確率は観測による更新が行われていないため初期値 (0.5) をとっている。

図中右上部は現在 token が存在する place を灰色の背景で表している (waiting, approach などの place はもう一つ想定される対象物運搬タスクを表し、このタスクは waiting 状態に固定されている)。安全確保タスクの初期 place は UNKNOWN であり、対話タスクについては LOST である。左中部はロボット頭部の軌道を表す。水平軸と垂直軸はそれぞれ pan と tilt 角を表す。右下部は二つのカメラで得られた画像を示している。細かい断片領域は肌色抽出による顔領域の候補を表しており、右下の比較的大きく抽出された領域は顔として抽出された領域を表す。画像の下には、ロボット頭部の座標系からみた顔方向の角度に加えてステレオ視の計算により得られた顔の位置の二次元位置 (カメラ方向に沿った平面上の位置) を示している。

Fig. 11 に人との相互作用の流れを示す。最初、部屋には人がいない (a)。人が一人部屋に入り、ロボットの近くにある椅

Fig. 12 Probability density estimation profile of human-detection

子に座る (b). しばらく椅子に座った (c) あと, 人はロボットに近づく (d). Fig. 12 はこの過程での人の存在確率分布の推定を表す. 最初は, すべての格子には 0.5 の確率が設定されている. しばらくあと, ロボットが周囲を見回すことにより, 周囲に人がいないことが確認され (Fig. 12-(a)), 安全確保タスクについての token が SAFETY に遷移する. 人が部屋に入ってくるとき, 足音により人の存在が検知される (Fig. 12-(b)). 確率分布が変化することにより頭部の動きは影響され, ロボットは人の方向を向くことにより時間を割くようになる. 安全確保タスクの token はいったん UNKNOWN に移るが, 人の方向への観測行動を行った結果, SAFETY に遷移する. このとき, 対話タスクの token は LOST から FOUND に遷移する. 人がロボットに接近したとき, 確率分布は変化し, ロボット近傍の格子が高い存在確率をもつようになる. これにより, ロボットは人に「何か用ですか」という発話をして人がロボットの動きに気付かせるようにする. この間, ロボットの頭部運動は生成された軌道候補から最適なものを選択され, 人のいない状況では定期的に周囲を観測する行動, 人が近くにいる状況では主に人のほうに注目しながら時々周囲を観測する行動, といった行動が実現された. これにより, 最適化の枠組みの下で人と相互作用するタスクにおける適切な首振り (観測) 行動の設計が可能であることを確認した.

4.2 動き回る人による評価

提案調停法はシミュレーションおよび実機実験において, ロボット周辺を人が歩きまわる環境で評価を行った. 評価の対象は, 式 (7) で表される最適化の枠組みである. しかし, 最適化の枠組み全体 (目標 place への到達時間など) を評価するのは人間の確率的な振る舞いを含むため困難である. そのため, 評価の対象から人の確率的な振る舞いを除き,

- 人の発話確率を含めて評価する代わりにロボットがどの程度人に働きかけたか (どの程度人を追跡したか) を評価する
- 人が任意に動き回る状況ではなく人の動く軌道を固定して評価する

という方法をとる.

シミュレーション実験

Fig. 13 に示すような軌道上を 2 人の人が歩きまわる. 図中の円の上では人は 3~8 秒間滞在する. 軌道の終端に達したときは, 軌道の初期位置に移動し, 制限時間内で軌道上を動き続けるものとする. それぞれの戦略について 10 回の試行を行い, 各試

Fig. 13 Trajectories of humans

行は 300 [s] とする. 各戦略は, 重みづけパラメータ w_1 および w_3 を変更することで生成する. 評価には, 注視時間, 看過時間, 平均未知格子数の三つを用いた. 注視時間は, ロボット頭部が人間の方向を向いていた時間を意味する. 注視時間は人の発話確率に影響を与えるため, 対話タスクの達成に寄与する時間である. 看過時間とは, 人がロボットの近傍領域 $R_{vicinity}$ にいるにもかかわらずロボットがそこにいる人を認識していない時間を意味する. この時間が長いほど, 安全確保タスクは十分に達成されていないとみることができる. UNKNOWN place にある token はできるだけ早く ATTENTION place に遷移することが求められるからである. 平均未知格子数とは, $p_{th1} < p_h < p_{th2}$ となる格子の数の平均値を算出したものである. この値は, ロボットが人の追跡よりも周辺の監視を重視した首振り行動をとるほど小さくなると考えられる.

注視時間, 看過時間, 平均未知格子数を異なる重みパラメータの組 $[w_1, w_3] = [1, 0], [1, 1], [0.7, 1], [0.5, 1], [0, 1]$ の戦略について評価した結果を Fig. 14, Fig. 15 および Fig. 16 に示す. 重みパラメータを $[w_1, w_3] = [1, 0]$ としたとき, 式 (7) の最適化は安全確保タスクのみを考慮して行われる. その結果として, ロボットは自身周辺を見回す行動をとる. 逆に, $[w_1, w_3] = [0, 1]$ としたとき, 対話タスクが最適化の際考慮される. この場合は, 人を見失わない限りは, ロボットは人を追跡するような行動をとる. 重みパラメータを変えることにより, 周辺の監視と人の追跡という 2 種類の観測行動の間のバランスをとるような調節が可能になる.

Fig. 14~Fig. 16 から, パラメータ w_1, w_3 を変更することでロボットの振る舞いが調節できていることが分かる. Fig. 14 では, 注視時間は $[w_1, w_3] = [0, 1]$ のときに最長である. つまり,

Fig. 14 Gazing time with different pairs of weights

Fig. 17 Gazing time in experiment

Fig. 15 Neglected time with different pairs of weights

Fig. 18 Neglected time in experiment

Fig. 16 Average number of unknown grids

重みパラメータの設定により対話タスクを優先することにより、対話タスクについての最良の性能が得られていることが分かる。逆に Fig. 16 では、平均未知格子数は $[w_1, w_3] = [0, 1]$ としたときに最大となっている。これは、周辺を監視して未知の格子を少なくするという安全確保タスクの性能が、人を追跡する行動を優先した結果として犠牲にされたということを意味している。Fig. 15 では、重みパラメータと看過時間の関係は平均未知格子数の Fig. 16 の結果と比較するとそれほど鮮明ではない。これは、看過時間の場合は確率的に決定される頭部の軌道と人の振る舞いの関係に影響されるためであると考えられる。

実機実験

重みパラメータの調節によりロボットの振る舞いが調節できることを実機ロボットにおいても検証するため、RI-MAN による実験を行った。2 人の人がロボットの周辺の決められた軌道上を歩いたり止まったりを繰り返しながら歩く。パラメータの組 $[w_1, w_3]$ に対する 1 回の試行は 420 [s] であり、その間人は軌道上を往復し続ける。ただし、人はロボットが顔を抽出・認識できるようにロボットの方向に顔を向け続けながら動き回るものとする。

注視時間を Fig. 17 に示す。注視時間は $[w_1, w_3] = [0, 1]$ の場合に最大であった。これは対話タスクに重みをつけ、安全確保タスクを無視した場合である。看過時間を Fig. 18 に示す。看過時間は $[w_1, w_3] = [1, 0]$ の場合に最小であった。これは、安全確保タスクに重みをつけ、対話タスクを無視した場合である。 $[w_1, w_3] = [0.7, 1]$ の場合、注視時間、看過時間両方の場合について中間的な結果が得られた。これより、シミュレーションで得られたのと同様の傾向の結果が得られていることが確認された。

5. 考察

4.1 節で述べた実験では、格子状の存在確率分布が変化することによって、最適化により得られる頭部の軌道も変化した。最初はロボットは周辺領域すべてを視野に収めるように活発に首を振ったが、後には、時々周辺を見回しながら近づいた人を注視し

た. このような振る舞いを実現するための一つの方法としては, 観測情報や状態に分岐条件を設けてその条件に合う場合に固定された行動を発現させるような if-then ルールを積み重ねる方法が考えられる. 例えば, 人が検知されたらその人を追跡する, 一定の時間が経過したら人の追跡をやめて周辺を見回す, などのルールが考えられる. このようなルールの合成によって本研究で扱っている振る舞いを実現することも可能であると考えられる. 事実, 本研究の枠組みで重みパラメータを $[w_1, w_3] = [0, 1]$ とした場合は「追跡首振り行動」とほぼ等価の振る舞いを行い, $[w_1, w_3] = [1, 0]$ とした場合には「周辺を見回す」に近い振る舞いを行う. しかし, if-then ルールを重ねる方法には, 行動を望ましいように設計する過程でルールを経験のみに基づいて試行錯誤的に調整・修正することで, 設計と実現される振る舞いの間の関係が分かりにくくなるという問題点がある. 提案法の一つの利点は, 人と接するロボットの振る舞いを設計・記述する上で設計者にしか分からない経験的な要素を少なくし, 何を最適化しているのかという見通しをよくすることである. 例えば, 提案法の改善する余地のある点の一つとして人の発話確率のモデル化を実験的に検証されたモデルに置き換えることがあげられる. このように, 行動を実現するための設計の枠組みの一部を置き換えたり改善したりすることができることは, 作りつけのルールによるアプローチでは困難であり, 本研究の別の利点であるといえる.

一方, 本研究で提案した人の挙動の確率モデル化と従来型のタスク設計手法を結びつけることもアプローチとしては可能である. しかし, この場合も複数タスク間の調停に明確な指針がないと, 設計者の経験に頼った調整になる. この意味で, 式 (8) を用いた調停の枠組みは必要である.

ロボットの全体の性能については, 重みパラメータの変更により調節する必要がある. 本研究で扱ったように相克する要求がある場合, 人とロボットの相互作用の具体的な結果に応じてパラメータを調節するという過程が求められる.

本研究の提案する枠組みの課題として, 現状では petri-net の利点の一つであるタスク間の同期や抑制といった並列動作の記述を定式化していない点があげられる. petri-net からマルコフ決定過程への変換について, 同期や抑制などの petri-net 上の遷移規則を考慮した定式化を導入できれば, 本研究で扱うことのできる対象を広げ, petri-net の持つソフトウェアによる整合性の自動チェック [26] などの利点を活用できると期待される.

6. おわりに

本研究では, 人と接するロボットのための, 不確実性を有する人の振る舞いを含んだ複数タスクの設計法として, petri-net を用いた定式化と最適制御・最適化の枠組みを用いた方法を提案した. 実機実験において, 人の振る舞いに合わせた周辺監視や人への注視の行動の切り替えが実現可能であることを確認した. また, シミュレーションおよび実機実験により, 提案設計方法のもとで重みパラメータを変えることによりロボットの振る舞いが調節可能であることが確認された. petri-net と最適制御の関係についてのより深い議論を行うことが今後の課題としてあげられる.

謝 辞 本研究において RI-MAN を用いた実機実験を行うにあたり, RI-MAN の研究開発に携わった理化学研究所バイオ・ミメティックコントロール研究センター研究員の諸氏の協力を得た. 特に, 実験時補助として平野慎也氏の協力を得た. ここに謝意を表する.

参 考 文 献

- [1] T. Kanda, T. Hirano, D. Eaton and H. Ishiguro: "Interactive robots as social partners and peer tutors for children: A field trial," *Human Computer Interaction*, vol.19, no.1-2, pp.61-84, 2004.
- [2] 神田崇行, 平野貴幸, ダニエル イートン, 石黒浩: "日常生活の場で長期相互作用する人間型対話ロボット", *日本ロボット学会誌*, vol.22, no.5, pp.636-647, 2004.
- [3] M. Shiomi, T. Kanda, H. Ishiguro and N. Hagita: "Interactive humanoid robots for a science museum," *IEEE Intelligent Systems*, vol.22, no.2, pp.25-32, 2007.
- [4] G. Kim, W. Chung, S. Park and M. Kim: "Experimental research of navigation behavior selection using generalized stochastic petri nets(gspn) for a tour-guide robot," *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005.
- [5] C. Breazeal: *Designing Sociable Robots*. MIT Press, 2002.
- [6] L.P. Kaelbling, M.L. Littman and A.R. Cassandra: "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol.101, pp.99-134, 1998.
- [7] 文仁赫, 三浦純, 白井良明: "不確かさを考慮した観測位置と移動のオンライン計画手法", *日本ロボット学会誌*, vol.17, no.8, pp.1107-1113, 1999.
- [8] J. Bazewicz: *Scheduling computer and manufacturing processes*. Springer, 1996.
- [9] R.A. Brooks: "A robust layered control system for a mobile robot," *IEEE Journal of Robotics and Automation*, vol. RA-2, pp.253-262, 1986.
- [10] J.H. Connell: "Sss: A hybrid architecture applied to robot navigation," *Proc. of the 1992 IEEE Conf. on Robotics and Automation*, pp.2719-2724, 1992.
- [11] E. Alami, R. Chatila, S. Fleury, M. Ghallab and F. Ingrand: "An architecture for autonomy," *International Journal of Robotics Research*, 1998.
- [12] A. Lehmann, R. Mikut and T. Asfour: "Petri nets for task supervision in humanoid robots," *Proc. 37th International Symposium on Robotics*, pp.71-73, 2006.
- [13] K. Kobayashi, K. Inoue and T. Ushio: "Llp supervisory control with time petri net models in mobile robots," *Proc. of the 2001 International Conference on Systems, Man & Cybernetics*, pp.3229-3234, 2001.
- [14] K. Kobayashi, A. Nakatani, H. Takahashi and T. Ushio: "Motion planning for humanoid robots using timed petri net and modular state net," *Proc. of the 2002 International Conference on Systems, Man & Cybernetics*, pp.334-339, 2002.
- [15] P.J. Haas and S.P. Nets: *Springer Series in Operations Research*. 2002.
- [16] A. Puliafito, M. Telek and K. S. Trivedi: "The Evolution of Stochastic Petri Nets," *Proc. World Congress on Systems Simulation* pp.3-15, 1997.
- [17] 小田島正, 大西正輝, 田原健二, 向井利春, 平野慎也, 羅志偉, 細江繁幸: "抱え上げ動作による移乗作業を目的とした介護支援ロボット研究用プラットフォーム "RI-MAN" の開発と評価", *日本ロボット学会誌*, vol.25, no.4, pp.554-565, 2007.
- [18] C.G. Cassandras and S. LaFortune: *Introduction to Discrete Event Systems*. Springer, 1999.
- [19] P.J.G. Ramadge and W.M. Wonham: "The control of discrete event system," *Proc. IEEE*, vol.77, no.1, pp.81-98, 1989.
- [20] B. Hruz and M.C. Zhou: *Modeling and Control of Discrete-*

event Dynamic Systems. Springer, 2007.

- [21] R.S. Sutton and A.G. Barto: 強化学習. 森北出版, 2000.
- [22] D.P. Bertsekas: Dynamic Programming and Optimal Control. Athena Scientific, 2005.
- [23] A. Elfes: "Using Occupancy Grids for Mobile Robot Perception and Navigation," Computer, vol.22, no.6, pp.46-57, 1989.
- [24] P. Stepan, M. Kulich and L. Preucil: "Robust data fusion with occupancy grid," IEEE Trans. on Systems, Man, and Cyber-

netics Part C, vol.35, no.1, 2005.

- [25] 中島弘道, 大西昇, 向井利春: "スペクトルの特徴マップを用いた上下方向音源定位学習システム", 電気情報通信学会論文誌 D-II, vol.J87, no.11, pp.2034-2044, 2004.
- [26] R. German, C. Kelling, A. Zimmermann and G. Hommel: "Timenet: a toolkit for evaluating non-markovian stochastic petri nets," J. Performance Evaluation, vol.24, pp.69-87, 1995.

小林祐一 (Yuichi Kobayashi)

1997 年東京大学工学部精密機械工学科卒業, 1999 年同大学大学院工学系研究科精密機械工学専攻修了. 2002 年同博士課程修了, 博士 (工学). 同年理化学研究所バイオ・ミメティックコントロール研究センター研究員. 2007 年より東京農工大学共生科学技術院特任准教授, 理化学研究所客員研究員. ロボット学習と自律分散システムの研究に従事. (日本ロボット学会正会員)

細江繁幸 (Shigeyuki Hosoe)

1967 年名古屋大学大学院工学研究科博士前期課程修了 (金属工学専攻). 同年名古屋大学工学部助手. 同大学講師, 助教授を経て 1988 年名古屋大学工学部教授. 1994 年同大学大学院工学研究科教授. 2001 年より理化学研究所バイオ・ミメティックコントロール研究センターセンター長, 運動系システム制御理論研究チーム・チームリーダー (併任), 2007 年より理研-東海ゴム人間共存ロボット連携センター長, 現在に至る. 制御理論, システム理論などに関する研究に従事. 博士 (工学). (日本ロボット学会正会員)

大西正輝 (Masaki Onishi)

1997 年大阪府立大学工学部情報工学科卒業. 2002 年同大学大学院博士後期課程修了. 同年理化学研究所バイオ・ミメティックコントロール研究センター研究員を経て, 2006 年産業技術総合研究所情報技術研究部門研究員, 現在に至る. 理化学研究所客員研究員. ロボットの認知・知識処理に関する研究に従事. 電子情報通信学会, 電気学会, 映像情報メディア学会各会員. 工学博士. (日本ロボット学会正会員)

羅 志偉 (Zhiwei Luo)

1984 年中国華中工学院自動制御と計算機学部卒業. 同年中国蘇州大学教師, 1986 年愛知工業大学客員研究員. 1991 年名古屋大学大学院工学研究科情報工学専攻博士前期課程修了. 1992 年同大学大学院博士後期課程修了. 同年豊橋技術科学大学助手. 理化学研究所フロンティア研究員, 山形大学工学部助教授, 理化学研究所バイオ・ミメティックコントロール研究センター・環境適応ロボットシステム研究チーム・チームリーダーを経て, 現在, 神戸大学大学院工学研究科教授. ロボティクス, 制御工学の研究に従事. 計測自動制御学会, 日本神経回路学会, IEEE 等各会員. 博士 (工学). (日本ロボット学会正会員)