

パーソナルロボットのための音声対話インタフェース Speech Dialogue Interface for Personal Robot

開 一夫 佐藤 倫太 安西 祐一郎
Kazuo Hiraki Michita Satoh Yuichiro Anzai

慶応義塾大学 計算機科学専攻

Department of Computer Science, Keio University

E-mail: {hiraki, michita, anzai}@aa.cs.keio.ac.jp

概要

本研究では、パーソナルロボットと人間との柔軟インタラクションを実現するための音声対話システム Linta-II について述べる。Linta-II では、ロボットと人間を取り巻く外界の状況を考慮した対話処理を目的としている。特に本論文では、外界の状況に対応した自然な発話生成を行なうため注意機構を導入した。注意機構は、センサ情報を処理する注意エンティティのローカルな相互作用によって実現され、結果的に発話に必要な外界の情報がフォーカシングされることになる。本論文では、Linta-II で提案している注意機構の特性に関する評価を通して、本機構の有用性について述べる。

abstract

This paper describes Linta-II, a speech dialogue interface for human-robot interaction. Linta-II uses external information to develop flexible dialogue system. Particularly in this paper, we propose an attention mechanism for situated utterances. The attention mechanism focuses external information and the utterance module can generate situated utterances in terms of focused information. The paper evaluates the characteristic of the attention mechanism and shows its significance for speech dialogue interface.

1 はじめに

一般に、対話における発話文には、文脈や状況から自明なことは現れず、必要最小限の情報が表現される。つまり、発話文の解析や生成を行なうためには外界の情報が必要となる。このため、音声対話システムを構築する際に、外界の情報を獲得し発話処理に利するための機構が必要となる。外界の情報（非言語情報）を考慮した自然言語対話インタフェースの研究は今までに幾つか研究されてきている [1],[2],[3]。しかし、ここで問題となるのは、外界を認識するための処理コスト、および、上位のモジュールが認識結果を参照するコストが、センサ数が増えるとともに増加することにある。

本論文では、この問題を解決するための注意機構を提案する。注意機構を用いることにより、発話機構やプランナなどが処理に必要なセンサ情報を優先的に処理することができる。上位のモジュールがセンサ情報を参照するコストは、注意に伴うフォーカシングによって低減される。

注意機構の導入で注目すべき点は、あらかじめ状況に関する複雑なルールを用意することなく、フォーカシングによって状況依存的発話生成が行なえることである。Linta-II の注意機構は、発話生成だけでなく、ユーザの発話文の理解や動作プラン生成といった一般的なタスクに利用することができる。

現在、この注意機構は、我々が開発した自律移動ロボット Einstein-I 上に実装されており、注意機構を利用したパーソナルロボット用対話システム Linta-II の作成を行なっている。本論文では、実装された注意機構の特性を評価し、また、注意機構に伴うフォーカシングの影響で生成される状況依存的発話の例について取り上げる。

2 注意を用いた処理

Linta-II では、対話処理に必要な外界の情報を獲得する際に、情報のすべてを獲得するのではなく、発話機構やプランナが必要とする情報に注意することにより、外界の情報の一部を優先的に処理し獲得することを試みている。ある情報に注意をするとは、その情報を獲得するための処理を優先的に行なうことである。また、注意をすることに伴い、参照できるセンサ情報は絞り込まれることになる（フォーカシング）。

Linta-II における注意には以下の二つがある。

1. 意図的注意
2. 反射的注意

本節では、この二つの注意を中心に、注意を用いた処理について述べる。

2.1 意図的注意

意図的な注意とは、発話機構やプランナに代表されるアプリケーションプログラムが処理に必要としている情報を効率良く獲得するために、アプリケーションプログラム側からトップダウンに構成される注意である。例えば、Einstein-I が前進している時には、前に障害物がないか注意しなければならない。つまり、他の方向よりも、前のセンサデータを優先的に処理するのが望ましくなる。

2.2 反射的注意

Einstein-I が行動する世界は、実世界 (Real World) であり、アプリケーションプログラムが意図的に注意している情報の他に、Einstein-I の行動に必要な出来事が起きた場合素早く反応できる必要がある。反射的注意とは、このような意図していない出来事をセンサが捉えることにより、センサ側からボトムアップに構成される注意である。例えば、Einstein-I の横に物が落ちてきて、超音波距離センサの値が急に変化した時に、反射的注意によって、落ちてきた物を避けることや、状況の変化にすぐに対応して、「ビクリシタ」と、発話を行うことができる。

2.3 注意によるフォーカシング

センサ情報が注意されることにより、アプリケーションプログラム側から参照できるセンサ情報は変化することになる。つまり、アプリケーションプログラムが参照できる外界の情報は、Einstein-I が獲得することのできるセンサ情報の一部分に限定（フォーカシング）されるとになる。例えば、Einstein-I が前進している時には、前の距離センサから得られる情報が意図的に注意されているため、参照できる情報は、前の距離センサから得られる情報が多くなる。Einstein-I が止まっている時には、センサ情報が均等に注意されているため、どの方向のセンサ情報も等しく参照することが可能になる。

このフォーカシングが起きることにより、外界の状況に依存した発話が可能になる。例として、Einstein-I が、次に何の動作ができるか（できないか）をユーザに報告（「動作可能性報告」）する場面を考えよう。ここで、Einstein-I の前には壁があるとすると、Einstein-I が前進している時には、前のセンサ情報が参照される確率が高いので、Einstein-I は「そろそろ前に進めません」と発話することになる。Einstein-I が停止している時には、どの方向のセンサも等しい確率で見えるので、「右に行きます」など他の発話をするのが可能になる。

最初の例の場合、前進ができなくなることをユーザに伝えていたが、この発話の生成において、発話機構は前進していることを考慮に入れる必要がない。つまり、注意がによるフォーカ

スの影響で、状況に依存した発話が生成されたのである。この例のように、注意機構を導入することにより、「前進しているから、前進の動作可能性報告を行なう」といった記述をシステムにすることなしに、状況に依存した発話を可能することができる。

3 注意機構

注意機構は、図1に示すように、センサ情報を処理する手続きの単位である注意エンティティの集まりで構成される。各注意エンティティは、注意度と呼ばれる活性度を個別に持っており、注意度が高くなるに従ってより強く注意され、各エンティティに対応した属性の value を書き込むことによってセンサ情報を、発話機構やプランナといった上位のモジュールに提供する。

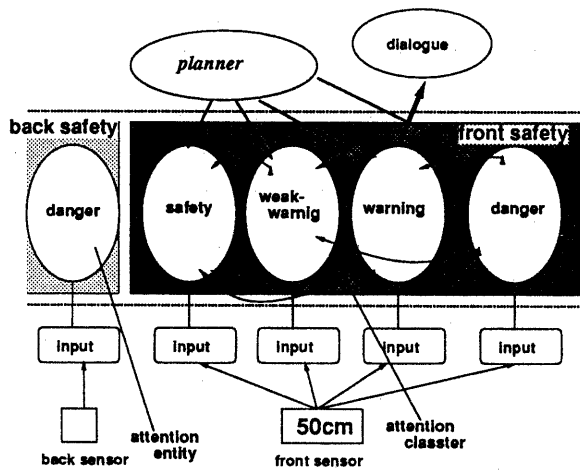


図 1: 注意機構の概略図

3.1 注意クラスと注意エンティティ

属性は、注意エンティティの集まりである注意クラスに対応する。図1では、属性 front safety は、濃い色の四角で囲われた四つの注意エンティティ(safety, weak-warning, warning, danger) の集まりに対応する。現在、注意クラスは四つ (front safety, back safety, right safety, left safety) 実装されている。属性の値は、注意クラス内のエンティティのローカルな相互作用 (次節で述べる) により決定される。

各注意エンティティは、各々の属性の値 (safety, weak-warning, warning, danger) に対応するものであり、センサデータ処理部 (sensory data processor) , 注意度計算部 (attention energy

calculator)、イベント処理部 (event processor) で構成される (図2)。

センサデータ処理部では、発話機構やプランナに必要な情報をセンサデータから処理し生成する。注意度計算部では、注意エンティティの注意度を計算する。現在の実装におけるセンサデータ処理部では、Einstein-I の持つ超音波距離センサから得られる距離情報から、Einstein-I が進む方向の安全性 direction safety の値 value (safety, weak-warning, warning, danger) を取り出している。各値は、超音波距離センサの値によって以下のよう に決まる。

- 0~15cm -> 危険 (denger)
- 16~100cm -> 注意 (warning)
- 101~200cm -> 要注意 (weak-warning)
- 201~250cm -> 安全 (safety)

注意エンティティは、この値の一つに各々対応している。

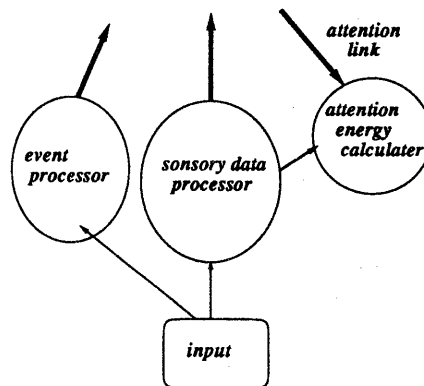


図 2: 注意エンティティ

イベント処理部では、外界の急激な変化に対処するための処理を行なう部分であり、反射的注意の構成を実現している。この処理は、以下の手順に沿って割り込みベースで行なわれる。

1. センサのデータの急激 (特徴的) な変化
2. 割り込みによるイベント処理部の起動
3. センサデータ処理と注意度再計算
4. 発話機構やプランナなどへ割り込み

現在の実装におけるセンサデータの変化は超音波距離センサから得られる距離の微分としている。

現在、各注意クラスは、センサの方向（前、右、左、後）に対応しており、4つあることになる。また、注意エンティティは、注意クラス内に4つ（safety, weak-warning, warning, danger）あり、全体で、16個あることになる。

3.2 注意度の決定

注意度関数

各注意エンティティの注意度は、以下の三点で決まる。

1. 意図的注意に関する注意度 (Top_at)
2. センサデータの処理の適合度合に関する注意度 ($Bottom_at$)
3. 他のエンティティとの相互作用から受ける抑制 ($Another_inhibi$)

注意度は、これらの値を引数に持つ注意度関数によって、各注意エンティティの注意度計算部により個別に計算される。注意度関数は下の式により表される。

$$Af(t) = Bottom_at e^{-time/(Top_at - \sum_j Another_inhibi_j)} + (Top_at - \sum_j Another_inhibi_j)$$

ここで、 $Another_inhibi_j$ は、注意エンティティ j より受ける抑制であることを示す。図3に、注意度関数のグラフを示す。この関数は、 $Bottom_at$ と Top_at の和を最大値として、時間がたつにつれ減衰する特徴を持つ。また、他のエンティティから抑制 $Another_inhibi$ を受け、最大値と時定数が減少し全体として注意度が減少する。

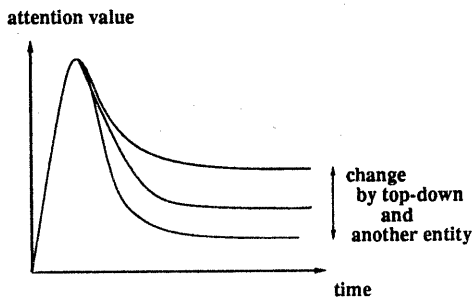


図 3: 注意度関数

意図的注意は、発話機構やプランナにより Top_at の値が高くなるに従って、注意度関数に変形し、注意が持続するようになることにより実現されている。図1では、前の安全性 (front safety) の情報を提供する注意エンティティの Top_at の値をプランナが高くなることにより、属性 front safety を持つ注意クラスがより強く注意されている。

相互作用による注意度の決定

同じ注意クラスに属する注意エンティティ間は抑制結合で結ばれており、処理の際にローカルな相互作用を起こす。属性の値は、センサデータの処理適合度に関する注意度 $Bottom_at$ を一番強く受けるエンティティが相互作用の際に強く注意されることにより決定される。

図1では、意図的に注意されている属性 front safety を持つ注意クラスにおいてローカルな相互作用がおき、センサデータ 50cm に対応する注意エンティティ warning が高い注意度を獲得し、属性 front safety の値 warning が決定している。

3.3 キューによる注意機構へのサポート

Einstein-I には、計算資源の有効利用のため6段階優先順位付キューが組み込まれている。Linta-II の注意機構では、注意度を優先順位と見なすことにより、注意度の高い注意エンティティが優先的に処理されるようになる。また、一つ活性化されると、他が相対的に抑制されることになる。

注意機構におけるフォーカシングは、トップキューと呼ばれるキューによって実現されている。処理されたセンサ情報は、トップキューに登録され、アプリケーションプログラムにより参照されることになる。トップキューは、一定の時間間隔で動くため、更新する回数が多い情報ほど多く登録されることになる。つまり、注意度の高いエンティティが多く登録されることになる。よって、トップキューには、注意度の高いエンティティがある時にはそのエンティティからの情報が多く登録され、各々のエンティティの注意度が均等の時には複数のエンティティが登録されることによりスコーピングは実現される。

4 発話機構

Linta-II の発話機構は、発話意図とセンサ情報から発話文を決定する以下の形をしたルールを持つ。

[発話意図][センサ情報] → 発話文

以下に、ルールの一部を例として幾つかあげる。

1. [動作可能性報告][right safety[safety]] → [右へ曲がれますが]

2. [動作可能性報告]

[*front_safety[warning]*]-> [そろそろ前に進めません]

3. [動作可能性報告]

[*right_safety[warning]*] -> [曲がれないのですが]

これらのルールは、上1. から順に参照され、最初に条件がマッチしたものを選んでいる。上のルールの場合、[動作可能性報告]であるため、「ロボットがある動作を実行可能である」ことを伝えるルールが先に来ており、「実行不可能である」ことを伝えるルールが後に来る。発話機構では、対話プランにより生成される発話意図とセンサ情報を参照することにより発話を生成する。

外界の情報に反射的な発話はセンサ情報により起動される。

5 注意機構を用いた状況依存的発話

Linta-II の発話機構では、注意機構により提供される外界の情報を参照して発話文を選択している。ここで、外界の状況に依存した発話の生成について、発話ルールと注意機構を用いて説明する。

Einstein-I が前進している時には、前に障害物が存在するかどうか *planner* により注意されるため、参照できる情報は属性 *front_safety* が多くなる。50cm 前に障害物があったときには、参照できる情報は *front_safety[warning]* となる。他の方向の安全性の情報はフォーカシングにより参照できないので、ルール [動作可能性報告][*front_safety[warning]*]-> [そろそろ前に進めません] が適用され、ユーザに前に進めないことを伝える。

Einstein-I が止まっている時には、均等にセンサ情報が注意されているため、どの方向のセンサ情報も等しい確率で参照することが可能になる。この場合、50cm 前に障害物があり他の方向に障害物がないときには、注意機構では *front_safety[warning]* 以外に *right_safety[safety]* を参照することができるので、ルール [動作可能性報告][*right_safety[safety]*]-> [右へ曲がれますが] が適用され、ユーザに右に曲がれることを伝える。

ここで用いた発話ルールは、状況依存的な記述（前進している時は、前のセンサを見よなどの記述）が予め用意されているわけではなく発話意図とセンサ情報のみによって決まる単純なものである。上記の例の状況依存的発話は、複雑なルールを用意することなく、注意機構によってつくり出されるフォーカスによって実現される。

注意機構が持つ反射的注意により、外界の動的変化に対応した発話も可能となる。例として、Einstein-I の右に物が落ちてきた場合について考える。この時、以下の手続きがとられる。

- 右の距離センサの値が急激に変化する。

- *right_safety* の注意クラスタに属する各エンティティのイベント処理部が割り込みによって起動される。

- 割り込まれた各エンティティは、センサデータ処理をし、注意度を再計算する。

- 発話機構へ割り込みを掛ける。

よって、起動された発話機構は、獲得された情報に応じた発話文を生成する。この場合は「ビックリシタ」という発話文を生成することになる。

6 実装と評価

6.1 実装

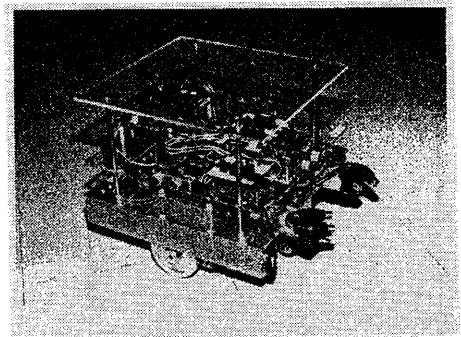


図4: 自律移動ロボット Einstein-I

Linta-II の注意機構は、我々の研究室で開発された自律移動ロボット Einstein-I (図4) に実装されている。Einstein-I は、0~250cm の範囲で物体との距離を計る超音波距離センサを前後左右に持つ。また、Einstein-I には、割り込みベースのプログラミングをサポートするロボット用 OS *pulser*[4] が乗っている。

6.2 評価

本論文では、以下の項目について実験を行ない、Einstein-I 上に実装された注意機構の処理の特性を調べた。

1. アプリケーションプログラムに意図的に注意されるセンサ情報が優先的に処理される過程を見る。注意機構では、Einstein-I が前進している時に、前面センサに対応する注意クラスタに意図的に注意しているとする。この際に、意図的注意の度合 *Top.at* が変化した時、注意されたクラスタが他のクラスタに対しどのくらい優先的に処理されるかについて評価をとる。

2. センサデータの処理の適合度に関する注意 $Bottom_at$ を受けるエンティティが相互作用により受ける特性について見る。Einstein-I の前方 50cm に障害物がある時を想定し、センサからボトムアップに得る注意度 $Bottom_at$ と意図的注意の度合 Top_at の各々の変化に対して以下の特性を調べる。

- (a) ローカルな相互作用により一番高い注意度を得るエンティティ（この場合、*front safety*[warning]）と他の全エンティティとの間の処理回数比について評価をとり、相互作用に勝ったエンティティが、どれだけ全体的に優先処理されるかを見る。
- (b) ローカルな相互作用において、センサ処理により注意度 $Bottom_at$ を得るエンティティと同じクラスタ内の他のエンティティとの間の処理回数比について評価をとり、 $Bottom_at$ が高いエンティティのクラスタ内の優先度について見る。

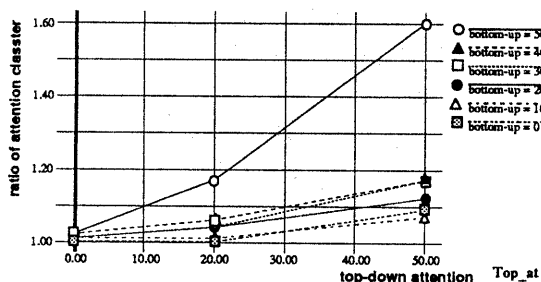


図 5: 意図的注意により注意されたクラスタと他のクラスタの処理回数の比

図 5 では、1. の意図的注意度 Top_at の変化に応じた、注意されたクラスタと他のクラスタの処理回数の比の変化を示したものである。

図 5 の横軸は意図的注意 Top_at の度合、縦軸は注意されているクラスタと他のクラスタとの比を表している。つまり、ここでは、*front safety* の注意クラスタが他の注意クラスタに比べてどのくらい優先的に処理されているか表現している。また、図のドットの種類は、センサ側からボトムアップに与えられる注意度 $Bottom_at$ の違いである。この図から、 Top_at が高い注意クラスタは、他に比べて処理回数が通常 1.1~1.2 倍増すことが分かる。つまり、Einstein-I が前進している時には、前の距離センサに対応する処理が優先的に行なわれ、停止している時には、均等に処理されることが分かる。また、ボトムアップに与えられる注意度が大きいほど注意されたクラスタはより優

先的に処理されていることがわかる。これは、ボトムアップに注意度を受けたエンティティが、クラスタ内の相互作用に勝ち、より優先的に処理されることを表す。

図 6 は、横軸がセンサ処理に関する注意度 $Bottom_at$ 、縦軸が相互作用に勝った注意エンティティの処理率を表し、ローカルな相互作用に勝った注意エンティティが全注意エンティティに対して優先的処理される度合 (2.a) が分かる。また、図中の変数 top は、意図的注意 Top_at を表す。この図より、センサ情報の処理の結果から得られる注意度 $Bottom_at$ に加えて、さらに意図的に注意されることにより、より優先的に処理されることが分かる。逆に、意図的に注意されていない場合には、たとえセンサ処理からえられる注意度 $Bottom_at$ が高くてもそれほど優先的に処理されないことが分かる。よって、Einstein-I が前進している時 50cm 前方に障害物があった場合、値 *warning* に対応する注意エンティティが優先的に処理され、*front safety*[warning] が得られることになる (*front safety* が意図的に注意されているため)。停止している場合には、*front safety*[warning] と他のセンサ情報処理とは均等に行なわれることが分かる。また、 $top = 50$ の時のグラフが階段上になっているのは、キューが 6 段階の離散的な優先順位を用いているためである。

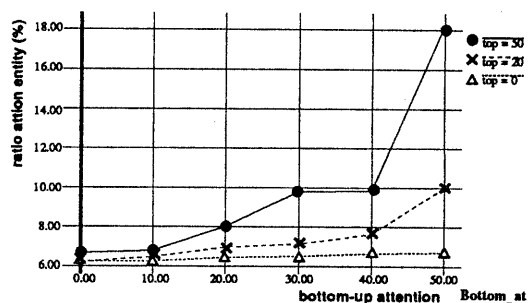


図 6: 相互作用に勝ったエンティティの処理率

図 7 は、横軸がセンサ処理により得る注意度 $Bottom_at$ 、縦軸が相互作用で勝ったエンティティとクラスタ内の他のエンティティとの処理回数比を表す。つまり、*front safety* の注意クラスタ内で起こる相互作用に勝ったエンティティと負けた他のエンティティとの処理回数比を表す。この図より、センサ処理に関する注意度 $Bottom_at$ を得たエンティティは、同じクラスタ内の他のエンティティと比べて、 $Bottom_at$ を得ることによりクラスタ内で優先的に処理される (2.b) とが分かる。つまり、注意エンティティは、センサ処理より注意度 $Bottom_at$ を得ることにより、クラスタ内の他のエンティティとの相互作用で勝つことができる。

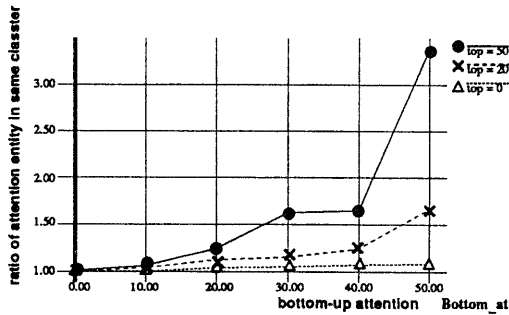


図7: 相互作用に勝ったエンティティと他のエンティティのクラスタ内における処理回数の比

7 おわりに

本論文では、発話機構やプランナに必要となるセンサ情報を優先的に処理し、提供する注意機構を提案した。注意機構は、センサ情報を処理する注意エンティティの集まりで構成され、各エンティティに対応した属性の値をトップキューに書き込むことによってセンサ情報を、発話機構やプランナといった上位のモジュールに提供するものとなっている。また、各属性の値は、注意エンティティ間のローカルな相互作用により決定される。

また、本論文では、自律移動ロボット Einstein-I に注意機構を実装し、注意機構の特性を調べた。注意機構の導入により、以下の処理が可能になった。

- アプリケーションプログラムにより意図的に注意されるセンサ情報の優先的処理
- 意図的注意以外に起きる外界の急激な変化に対する即時的センサ情報の処理
- 注意機構が注意するセンサ情報に対してフォーカスが構成されることによる簡単な状況依存的発話生成

注意機構によるフォーカシングを用いた状況依存的発話生成は、予め状況に関するルールを記述することなしに生成することが可能となっている。

現在 Litna-II で扱われている発話ルールの数は対話システムとしては少ない。今後は、注意機構をさらに有効利用した発話機構の開発が望まれる。また、対話システムへの注意機構の有効性を示すために、ユーザの発話文解釈への応用についても検討の必要がある。また、現在実装されているセンサは、超音波距離センサの一種のみであり、他の種類のセンサの利用を考

えた場合、センサ情報の属性間の関係（異なる注意クラスタ間の関係）をシステムに組み込む必要が生じるであろう。

References

- [1] 伊東, 高木, 六沢, 清水, 北岡, 小原: “二次元図形世界における日本語文と視覚情報の付き合わせ処理”, 信学論 (D), Vol. J69-D, pp.1083-1094, (1986).
- [2] J. G. Neal, Z. Dobes, K. E. Bettinger, J. S. Byoun: “Multi-Modal References in Human-Computer Dialogue”, AAAI-88, pp.819-823, (1988).
- [3] 佐藤, 開, 安西: “ロボットとの対話: センサ情報を利用した音声対話システム Linta の設計と実装”, 人工知能学会研究会資料, SIG-SLUD-9202-3, pp. 19-26, (1992).
- [4] 菅原, 矢向, 安西: “自律移動ロボット用 OS PULSER の設計と実装”, '92 情報処理学会 第 44 回 (平成 4 年前期) 全国大会 講演論文集, (1992).