

アクションプリミティブを用いた移動マニピュレータの知的動作の実現

— 本の検索・把持・運搬の動作設計 —

永谷 圭司 田中 豊

岡山大学 工学部

Intelligent Motion of Autonomous Mobile Manipulator based on Programming Style “Action Primitive” — Motion Design of Searching and Picking up Books —

Keiji Nagatani, Yutaka Tanaka

Department of Engineering, Okayama University

Abstract: Our research goal is to realize an intelligent autonomous motion for mobile manipulator. A model task, “Motion of Searching and Picking up Books”, was chosen as research task. To realize the task, the authors adopt a programming style named “action primitive” for robot controller. The action primitives are basic motion control programs for autonomous robots, and the robot motion is generated by executing a sequence of action primitives step by step. This paper describes motion design of above research task, and reports simulation results of mobile manipulator motion.

Key Words: Autonomous Mobile Manipulator, Action Primitive, Motion Planning

1 はじめに

1.1 研究の背景

近年、走行機能とマニピュレーション機能を同時に有する移動マニピュレータの研究が盛んに行われている。この移動マニピュレータは、広範囲で複雑な作業を行えるという利点を持つ反面、ロボットのベースの移動によって生ずるマニピュレータの先端位置誤差の問題や、マニピュレータ制御と走行の協調動作の問題など、解決すべき点がまだまだ多い。このため、現在のところ、移動マニピュレータに知的で複雑な作業を行わせるまでには至っていない。

一方、ロボットの活躍が期待される作業環境は、従来の工場内などの「ロボットのために整備された環境」から、オフィスビルや病院、家庭内といったロボットのために整備されていない「人が存在する環境」へと移行してきている。このような「人が存在する環境」では、予定外の状況に対処する能力や、自律的な判断を行う能力が、これまで以上ロボットに求められる。

1.2 研究目的

上記の背景をふまえ、本研究では、ロバストなロボットの知的動作の実現を目標とすることにした。さて、一概にロバストな知的動作といっても、目指す動作は漠然としており、研究方針が絞りにくい。そこで、「ある実現目標(タスク)を設定し、そのタスクを実現する過程でロボットに必要な能力を検討することにより、ロボットの知的動作の獲得を目指す」という研究手法を使用することとした(これをタスクオリエンテッドア

プローチと呼ぶ)。

本研究で設定したタスクは「本の検索・把持・運搬動作」である。ただし、対象とする環境としては、人が存在するオフィスや研究室などの屋内環境を想定する。

本稿では、このタスクを実現するための、ロボットの動作解析および動作設計について述べ、設定したタスクを実現するためのアクションプリミティブを用いたロボットの動作プログラミングについて報告する。

1.3 関連研究

移動マニピュレータを用いた応用研究は、国内外で盛んに行われている。その中でも、ドア開け動作に関する研究については、タスクの必要性和その実現の難しさから、様々なアプローチがなされてきた。

筆者は、1994年から1997年にかけて、タスクオリエンテッドアプローチに沿って、「自律移動ロボットが自分でドアを開閉して通り抜け、目的地まで走行する」というタスクを設定し、実環境においてこの動作を実現した [1]。

この他にも、日立製作所の岩本(関連研究は [2])、ミュンヘン工科大学の Schmidt(関連研究は [3])、電気通信大学の L.V. Litvintseva [4] らは、それぞれ独自のアルゴリズムを用いて「移動マニピュレータによるドア開け動作」という知的動作の設計および実装を行い、成果を収めた。また、スタンフォード大学の Khatib らは、複数の移動マニピュレータによる物資の協調搬送という高度な知的動作を目指した研究 [5] を行っている。また、東北大学の中野らは、メッセージを運搬するというタスクを実現する移動マニピュレータシステムを開発している [6]。

前述の通り、移動マニピュレータの応用に関する研究は盛んに行われているが、複雑な動作を組み合わせたロバストな知的動作を実現した例は、残念ながらまだ少ない。また、実現した動作の多くは、ロボットのために整備された特殊な環境の下で行われてきた。

そこで、本研究では、移動マニピュレータによる高度な知的動作を目指し、「本の検索・把持・運搬」という、複雑で、かつ将来的に移動マニピュレータで実現が期待されるタスクを設定した。

2 アクションプリミティブ

ロボットの知的動作は、様々なアクチュエータとセンサを、状況に応じて適確に組み合わせることによって初めて可能となる。したがって、ロボットの状況に応じた動作分割と遷移の枠組みを、明確に設定しておくことが望ましい。そこで筆者らは、上記の枠組みを考慮した“アクションプリミティブ”[7]というプログラミング手法を採用した。本章では、まずこのプログラミング手法に関する説明を行う。

2.1 動作の分割

本手法では、実現を目指す複雑な動作を解析して動作設計を行うため、まず目標とする全体の動作を時間に沿って分割し、基本動作の列を作成する。この分割の指標としては、

1. アクチュエータの種類の変更
2. センサフィードバック方法の変更

が挙げられる。ただし、ロバストな動作を実現するため、各基本動作は、センサフィードバックを用いた誤差対処能力を有する必要がある。このような、センサフィードバックを内包し、ロボットの各基本動作を実現するプログラムを「アクションプリミティブ」と呼ぶ。このアクションプリミティブは、上位の意志決定プログラムから動作パラメータと環境モデルを受け取り、リアルタイムにセンサ情報を入力しつつアクチュエータを制御する。これにより、ロボットの状態は、次の状態へ遷移する。

2.2 動作遂行の責任体制

複数の独立したプログラムが、同時に同一のアクチュエータを操作しようとした場合、首尾一貫した制御を行なうことはきわめて難しい。したがって、最終的にロボットの動作に責任を持つプログラム単位が必要である。ここではアクションプリミティブがこれに相当し、実行中のロボットの動作に全責任を持つ。このた

め、アクションプリミティブの実行時には、これが終了するまで、上位の意志決定プログラムからの干渉を受けない。

2.3 制御入力決定

動作パラメータは、アクションプリミティブ起動時に、上位の意思決定プログラムから受け渡される。アクチュエータへの制御入力は、この動作パラメータに基づいて、アクションプリミティブの実行開始時に計算される。

2.4 誤差への対処

ロボットのために整備されていない環境下では、ロボットが環境の完全なモデルを持つことは不可能であり、また、ロボットの動作自体にも必ず誤差が含まれる。この問題に対処するため、各アクションプリミティブは、センサより得られる環境の様子を常に観測し、計画動作と実動作の間の誤差をできる限り小さくするように常時制御入力を修正する。実動作がほぼ計画動作通り進んでいけば、計画動作と実動作の間の誤差は小さいはずである。そこで、推定した誤差を線形フィードバックすることにより、制御入力の修正を行う。

2.5 予定外の状況の監視

ロボットが、ほぼ計画動作通りに動作している時にも、走行経路上に障害物がある等、モデルと実環境との間に構造的相違が存在する場合がある。そこで、このような問題にも対処するため、アクションプリミティブは、自己の動作と環境の状況が計画の通りであるかを監視し、モデルと実環境との間に大きな相違が確認された時点で動作を中止する。その後、アクションプリミティブは、目的達成自体を中断しないために、アクションプリミティブの開始時の状態にロボットを戻し、意思決定プログラムにそれ以後の処理を任せることとする。

2.6 アクションプリミティブの終了

アクションプリミティブの終了には、以下に示す3つの状態が考えられる。

- (1) Success (動作完了) : 目的動作を達成
- (2) Fail (動作中止) : 初期状態に復帰
- (3) Give up (破滅) : 全ての動作を中止

これらの状態を意思決定プログラムに送り返した後、アクションプリミティブは動作を終了する。

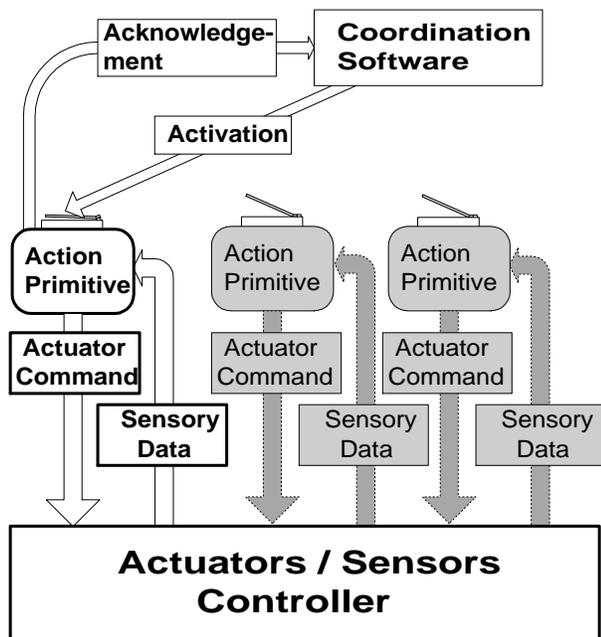


Fig. 1: Action Primitive Execution

2.7 意思決定プログラム

ロボット全体の動作は、適切なアクションプリミティブを順次実行することにより実現される。意思決定プログラムは、このアクションプリミティブのシーケンスを作成し、そのシーケンスにしたがって、アクションプリミティブを順次実行するものである (Fig.1 参照) .

このシーケンスを生成するため、状態を変化させる全てのアクションプリミティブを列挙し、各アクションプリミティブに、動作完了、並びに動作中止となった場合のアクションプリミティブの遷移先を付加したアクションプリミティブ遷移表を作成する。ロボットは、動作開始前に、このアクションプリミティブ遷移表の上でサーチを行うことにより、実行すべきアクションプリミティブのシーケンスを生成する。実際のロボットの実時間制御は、アクションプリミティブ実行系によって行なわれる。

さて、アクションプリミティブを実行中に予定外の状況が発生し、ロボットの状況 “Fail” が意思決定プログラムに返された場合、意思決定プログラムは、その状態を開始状態として、新たに別のアクションプリミティブのシーケンスを作成する。これにより、アクションプリミティブの動作が中止された状況への対処を行うことができる。

2.8 本手法の特徴

本プログラミング手法では、意思決定プログラムは、アクションプリミティブのシーケンスを生成し、これ

らを順次起動するという役割しか持たない。誤差への対処やロボットの状態監視、および予定外の動作中止からの回復などは、全てアクションプリミティブが行う。仮に、アクションプリミティブの動作中止がなければ、意志決定プログラムは単に順番にアクションプリミティブを起動するだけのプログラムとなる。しかし、一度アクションプリミティブの動作が中止された時、意志決定プログラムは動作手順を組み替えるという重要な役割をなす。このように、下位の実行系の責任の比重を大きくとり、上位のプログラムの担当を動作計画と再計画のみに限定する点が、本手法の大きな特徴である。これにより、従来型のアーキテクチャ (例えば、上位レベルのプロセスが下位のリアクティブなプロセスを制御するアーキテクチャ [8]) の不得意とする「予定外の状況への対処」が容易に実現される。

3 アクションプリミティブを用いた動作設計

3.1 タスクの解析

通常、我々人間が行っている「本を探し出し、取り出して運ぶ」という動作を考慮すると、「本の検索・把持・運搬」というタスクを実現するためには、ロボットに移動機構、マニピュレータ、およびロボットハンドが必須である。そこで、本研究で使用するロボットは、ロボットハンドを有する移動マニピュレータとする。

また、本研究では、「複数の本棚の位置は既知であるが、目的の本がどこの本棚にあるかは不確かである」という条件を付加した。このためロボットは、探索する本棚に目標とする本が存在しない場合、別の本棚へ移動し、そこで再び本を検索するという予定外の状況への対処動作が必要となる。

3.2 動作分割とアクションプリミティブ遷移表の作成

人間の動作をもとにして設計した、目標タスクを実現するためのアクションプリミティブ遷移表を、Table.1 に示す。この表中の、各基本動作を実現するプログラムが、本研究におけるアクションプリミティブに相当する。ただし、(0)、(7)、(8)、(10) については、ロボットの動作が直接伴わないため、アクションプリミティブとは言えないが、アクションプリミティブ遷移表に明記した。特に問題が生じない場合には、本表の (0) から (7) を順次実行することで、目標タスクは実現される。

	Name of Action Primitive	Parameter	Next (OK)	Next (Fail)
(0)	START		1	1
(1)	MOVE_TO	TARGET_POS	2	1
(2)	SEARCH	TARGET	3	10
(3)	APPROACH	TARGET	4	2
(4)	PICK_OUT	TARGET	5	3
(5)	MOVE_TO	GOAL	6	5
(6)	RELEASE	TARGET	7	9
(7)	GOAL			
(8)	GIVEUP			
(9)	MOVE_TO	START	8	8
(10)	CHANGE	TARGET		

Table. 1: Procedure of Search and Pick up Motion

3.3 アクションプリミティブの動作設計

3.3.1 経路追従走行 (MOVE_TO)

「経路追従走行」の目的は、移動ロボットの走行系を制御し、目標の本が保管されている本棚の前、または目的地まで走行することである。ここでは、動作開始前に、意志決定プログラムより与えられた環境モデル、現在位置、および目標位置より経路計画を行う。動作開始後、計画した経路に移動ロボットを追従させるように、実時間で走行制御を行うことで、経路追従走行を実現する。

さて、現在ある車輪型移動ロボットの多くは、オドメトリを用いて自己位置を推定する機能を有するが、この推定には、車輪のスリップや路面の凹凸によって生ずる位置誤差が累積するという性質がある。この問題に対処するため、本アクションプリミティブでは、環境中のランドマークが存在する地点で、環境モデルと比較し、外界センサ（視覚センサ）を用いたロボットの推定位置の修正を行う。

3.3.2 本の検索 (SEARCH)

「本の検索」の目的は、ロボットに搭載した視覚センサを用いて、目標の本の位置を検索・確認することである。

本の検索を行うためには、視覚センサが必要となるが、その際、カメラの視野角の狭さを補う必要がある。そこで、マニピュレータの先端に視覚センサを搭載する「ハンドアイ」を使用し、このマニピュレータを制御することによって、広範囲の検索を実現することにした。また、対象とする本の判別には、テンプレートマッチングが有効であると考えられる。よって、このアクションプリミティブが必要とする動作パラメータは、本を検索するためのマニピュレータの動作領域、および目的の本の画像テンプレートである。

3.3.3 ハンドのアプローチ (APPROACH)

「ハンドのアプローチ」の目的は、目標の本を把持するために、検索した本の近傍にロボットハンドをアプローチさせることである。

ロボットが推定する自己位置には誤差が存在するため、本棚に対し、ロボットが計画通りの位置に停止しているとは限らない。そのため、ロボットハンドにも、位置誤差が生ずる可能性が大きい。そこで、この誤差を減少させるため、ロボットハンドに搭載した視覚センサから得られる対象物の相対位置情報をフィードバックし、ロボットハンドの軌跡を修正する。

3.3.4 本の把持と取り出し (PICK_OUT)

「本の把持と取り出し」の目的は、目標の本の把持である。この動作は、通常のスライド式の2指のロボットハンドでは実現が不可能であると考えられるため、複雑で器用な動作を可能とする複数本の指を有するロボットハンドを使用する必要がある。ロボットハンドの動作は、予めオフラインで計画した各指の動作をプレイバックすることで実現する。

3.3.5 本のリリース (RELEASE)

「本のリリース」の目的は、ロボットハンドを制御して本を置くことである。この動作は、予めマニピュレータおよびハンドに動作列をティーチングすることで実現できると考えられる。

4 実装状況とシミュレーション

現在、筆者の所属する研究室では、設計した動作を実ロボット上で動作させるための、ロボットの整備を進めている。これと並行し、アクションプリミティブの動作をコンピュータ上で確認するため、移動マニピュレータシミュレータの開発も行っている。本章では、各アクションプリミティブの実装状況と、開発中の移動マニピュレータシミュレータについて紹介する。

4.1 ロボットプラットフォーム

本研究で使用するベースロボットには、筆者の所属する研究室が有する、全方向走行車を使用する。この走行車は、メカナムホイールという特殊な車輪を用いて、全方向移動を実現している [9]。一方、搭載するマニピュレータについては、現在設計段階である。

外界センサとしては、ロボットハンドの先端に搭載する視覚センサと、ベースロボットに搭載する視覚センサを使用する予定である。

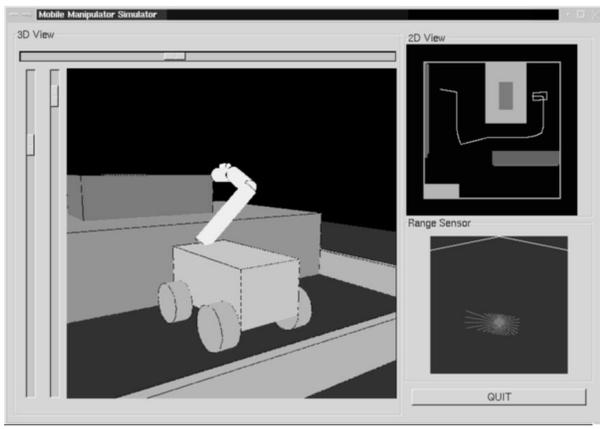


Fig. 2: Mobile Manipulator Simulator

4.2 シミュレータ

アクションプリミティブの遷移状況や、走行状況を確認するため、本研究では、移動マニピュレータシミュレータを開発した。外観を Fig.2 に示す。

このシミュレータの特徴は、ユーザプログラムとシミュレータとのインタフェースが、ユーザプログラムと実ロボットのインタフェースと同じであるという点である。したがって、シミュレータ上で動作するユーザプログラムは、実ロボット上でも動作が可能となる。また、本シミュレータは、マニピュレータの各目標関節角を指定することで、マニピュレータのキネマティクスを計算し、ベースロボットとマニピュレータの動きを3次元的にコンピュータ画面上に表示できる。さらに、ロボットがレンジセンサを搭載していることを仮定し、物体までの理想距離を獲得して、ユーザプログラムに提供することができる。

しかしながら、本シミュレータでは、マニピュレータや走行系に加わる力や動特性を考慮しておらず、また手先に搭載する視覚センサの情報も考慮していない。このため、本シミュレータの用途は、あくまでアクションプリミティブの遷移状況の確認と、ユーザプログラムの動作の確認である。

4.3 経路追従走行 (MOVE_TO) の実装

経路追従走行では、まず走行経路を計画する必要がある。対象とする環境が既知であるため、ここではロボットの経路計画に、Reduced Generalized Voronoi Graph (RGVG) [10] を使用することにした。

2次元平面上の Generalized Voronoi Graph (以下 GVG と表記する) は、

$$G(x) = [(d_1 - d_2)](x) = 0 \quad (1)$$

で表現される [11]。ここで、 d_1 と d_2 は、2つの凸形状

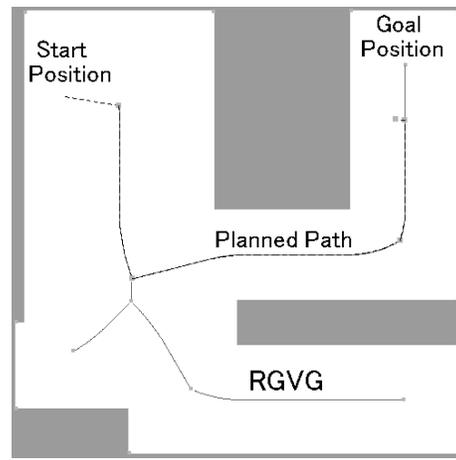


Fig. 3: Reduced Generalized Voronoi Graph

の障害物までの距離値である。この GVG から、環境に接する不要なエッジを削除したものが Reduced GVG であり (以下 RGVG と略す)、本研究では、この RGVG を用いて、以下の手順で経路計画を行う。

1. スタート地点と、そこから最短距離にある RGVG 上の点を結ぶ (Accessing RGVG)
2. 目的地と、そこから最短距離にある RGVG 上の点を結ぶ (Departing RGVG)
3. グラフサーチにより、上記の2つの RGVG 上の点を結ぶ最短経路を求める (Traversing RGVG)

RGVG を用いた経路計画例を Fig.3 に示す。

計画した経路は、短い線分の集合で表される。本研究では、順次、この線分へのベースロボットの追従制御を行うことで、目標経路追従走行を実現することとした。

現在、上記の経路追従動作がシミュレータ上で動作しており、Fig. 2 はその一例である (この図は、ベースロボットが、計画した経路を走行し、目的地に到達したところである)。ただし、本実装では、推定位置に生ずる誤差は考慮していない。

4.4 ハンドのアプローチ (APPROACH) の実装状況

現在、筆者らは、移動ロボットに搭載可能な小型軽量マニピュレータを設計中である。また、このマニピュレータを制御するための、ISA バスを用いた DC モータ制御ボードを試作中である。

そのため、現時点では、ハンドをアプローチさせるマニピュレータの動作計画を行い、シミュレータ上でアプローチ動作を実現している。

4.5 意志決定プログラムの実装

現在,実装されているアクションプリミティブは,経路追従走行,およびハンドのアプローチである.また,その他のアクションプリミティブには,何の動作も行わないダミープログラムを挿入した.この状態で,これらを統括する意志決定プログラムを実装し,シミュレータ上で動作確認を行った.ただし「本の検索」(SEARCH)では,本が検出できたかどうかの判断入力を,ユーザがキーボードより与えるものとした.

実装した意志決定プログラムを実行すると,ロボットは走行経路を計画し,最初の本棚の前で停止する.ここで,本が検出できた場合(オペレータがキーボードよりその旨を入力),ロボットはゴールへ向かう.一方,本が検出できなかった場合,意志決定プログラムは,別の本棚を目指すためのアクションプリミティブを生成し,動作を続行する.このことから,予定外の状況により一連の動作が中断されても,本プログラミング手法を使用することで,全体の動作を中止することなく動作を続行できることが確認できた.

5 まとめと今後の課題

本稿では,目標タスク「本の検索・把持・運搬動作」を実現するための,アクションプリミティブを用いた移動マニピュレータの動作設計を行った.また,経路追従走行とハンドのアプローチのアクションプリミティブ,および意志決定プログラムを実装して,これらの動作確認をシミュレータ上で行い,予定外の状況にも対処できることを確認した.

実環境においてこの動作を実現するためには,ロボットのベースが移動することで生ずるマニピュレータの先端位置誤差の問題や,マニピュレータ制御と走行の協調動作の問題など,解決すべき点が多い.しかし,具体的な問題全てを現在の段階で発見することは不可能である.したがって,今後設定した基本動作を一つ一つ実現する過程で問題点を洗い出し,これを克服することで,目標タスクの実現を目指したい.

謝辞

工学博士 油田信一教授(筑波大学)には,アクションプリミティブの考案に際しご指導を頂いた.ここに感謝の意を表す.

参考文献

[1] 永谷圭司, 油田信一. タスクオリエンテッドアプローチによる自律移動マニピュレータの研究—ドアの通り抜けを含む屋内の自律走行の実現—. 日本ロボット学会誌, Vol. 17, No. 6, pp. 111–121, 1999.

- [2] 岩本太郎, 山本広志, 榊原義宏. 複合センシングによる屋内通路の環境把握と形状可変クローラの自律移動制御実験. 日本ロボット学会誌, Vol. 6, No. 5, pp. 405–415, 1988.
- [3] Kianoush Azarm and Gunther Schmidt. Integrated mobile robot motion planning and execution in changing indoor environments. In *Proc. of IEEE/RSJ Int. Conference on Intelligent Robots and Systems*, pp. 298–303, 1994.
- [4] L.V.Litvintseva, T.Tanaka, T.Hirabayashi, Y.Watanabe, K.Ogino, M.Hamuro, K.Saeki, K.Yasukawa, and S.V.Ulyanov. Intelligent mobile robot for service use in office buildings developed through university-industry cooperation (uic). 第3回ロボットティクスシンポジウム予稿集, pp. 197–202, 1998.
- [5] Oussama Khatib. Robot planning and control. In *Proc. of International Workshop on Some Critical Issues in Robotics*, pp. 65–80, 1995.
- [6] 佐々木裕之, 高橋隆行, 中野栄二. 前方向移動ロボットに搭載されたマニピュレータによるドア開け動作の研究. 第13回日本ロボット学会学術講演会予稿集, pp. 711–712, 1995.
- [7] 油田信一, 永谷圭司. アクションプリミティブを単位とした自律ロボットの動作設計と制御—不整備環境に対処するためのプログラミング—. 第15回日本ロボット学会学術講演会予稿集 Vol.2, pp. 477–478, 1997.
- [8] R.C.Arkin. Integrating behavioral, perceptual, and world knowledge in reactive navigation. *Journal of Robotics and Autonomous Systems*, Vol. 6, pp. 105–122, 1990.
- [9] M. Blackwell. The uranus mobile robot. In *Carnegie Mellon University Technical Report CMU-RI-TR-91-06*, 1990.
- [10] Keiji Nagatani and Howie Choset. Toward robust sensor based exploration by constructing reduced generalized voronoi graph. In *Proc. of IEEE/RSJ Int. Conference on Intelligent Robots and Systems*, pp. 1687–1692, 1999.
- [11] H. Choset and J.W. Burdick. Sensor Based Planning, Part II: Incremental Construction of the Generalized Voronoi Graph. In *Proc. IEEE Int. Conf. on Robotics and Automation*, Nagoya, Japan, 1995.