

# 移動ロボット位置決めのためのセンサプランニング -ベイジアンネットワークの表現と構造学習- Sensor Planning for Mobile Robot Localization -Use of Bayesian Network Representation and Structure Learning-

周 洪鈞\*  
中央大学大学院

坂根 茂幸†  
中央大学理工学部

**Abstract:** 自律ロボットの構成において、不確実性を含む状況下で自己位置を推論する技術は重要である。本論文は、移動ロボットの不確実性を含む位置決め問題に関するセンサプランニング手法を提案する。具体的には、ローカルなセンサ情報と行動とグローバルな位置決めとの関係をベイジアンネットワーク (BN) で表現する。システムは初めに、ノードの順序付けを探索する GA と結合した K2 アルゴリズムを用いて BN の構造を学習する。実行時には、学習した BN に基づき、グローバルな位置決め信念度とセンシングコストのバランスを考慮して能動的なセンシング行動を計画する。シミュレーション実験により本システムの有効性を検証した。

## 1 はじめに

自律ロボットの構成において、不確実性を含む状況下で自己位置を推論する技術は重要である。本論文は、移動ロボットの不確実性を含む位置決め問題に関するセンサプランニング手法を提案する。具体的には、ローカルなセンサ情報と行動とグローバルな位置決めとの関係をベイジアンネットワーク (BN) [1] で表現する。システムは初めに、ノードの順序付けを探索する GA と結合した K2 アルゴリズムを用いて BN の構造を学習する。また、環境データに基づき CPT(条件付確率テーブル)のパラメータ値も学習する。実行時には、学習した BN に基づき、グローバルな位置決め信念度とセンシングコストのバランスを考慮して能動的なセンシング行動を計画する。シミュレーション実験により本システムの有効性を検証した。

## 2 従来の研究

ベイジアンアプローチで移動ロボットの誘導と位置決めをする方法はこれまでも提案されてきた。Thrun[7]ら

は、パーティクルフィルタ (Particle Filter) を用いた移動ロボット位置決め方法を提案した。ロボットは、位置決め信念度 (belief) をパーティクル集合で近似した事後確率分布で表し、ロボットの移動とセンサによる観測のモデルに基づいて逐次更新を行う。しかし移動ロボットの行動はランダムでありセンサプランニングは行っていない。そのためパーティクル集合の収束効率が悪い場合も起こり得る。

環境情報とロボットの行動を静的な BN や動的な BN でモデリングする研究も行われた。Asoh[10]らは予め構築した BN を用いて音声会話での応答を evidence として BN に入力しロボット位置推定の曖昧さを減少させている。しかし、センサプランニングの機能は含まれていない。Basyeら [13]のシステムは、動的な BN (DBN) を使ってセンサ情報を融合し、逐次行動を決定した。

移動ロボットの誘導と位置決め分野では、確率計算に基づくセンサプランニングの研究が行われている。Kristensen[16]は、決定木を用いてセンシング行動を計画した。システムは各プランニングのステップで可能なセンサを選択しているため計算は複雑になる。しかも、ユーティリティを考慮した決定木は簡単すぎるので、センサ情報の間の因果関係やセンサ情報と行動の間の因果関係を表現することは難しい。Miuraら [11]は、簡単なベイジアン推定に基づき、センシングコストとセンシング

\*〒 112-8551 東京都文京区春日 1-13-27、中央大学大学院理工学研究科 経営システム工学専攻, zhou@indsys.chuo-u.ac.jp, URL: <http://www.indsys.chuo-u.ac.jp/~zhou>

†〒 112-8551 東京都文京区春日 1-13-27、中央大学理工学部 経営システム工学科, sakane@indsys.chuo-u.ac.jp, URL: <http://www.indsys.chuo-u.ac.jp/~sakane/sakane.html>

不確かさのバランスを考慮して移動ロボットの行動計画を行った。一方、ビジョン分野の研究では、RimeyらがBNを用いて能動視覚の行動計画を提案し、テーブルセッティングの認識における next view problem を扱った試みがある [12]。

近年、データから最適なBN構造を学習する研究 [15] [3] [4] が、理論のみならず現実問題への応用 [5] も含めて発展しつつある。しかし、BN構造学習の移動ロボットへの応用、特にセンサプランニングへの応用については未開拓である。

### 3 移動ロボットの位置決め問題

移動ロボットの位置決め問題には、一般に以下のようなクラスがある [7] :

1. position tracking
2. global localization problem
3. kidnapped robot problem

本研究では、この中で最も難しいとされる 3. の kidnapped robot problem を扱っている。(例えて言えば) ロボットに目隠しをして別の場所に移動した後に目隠しを取り外す、という状況である。ロボットは、そこから動き出してセンシングを行い、自己位置を推定する。

本センサプランニングのシステムは、まず、オフラインで、(a) 環境データの収集と (b) BN による環境の表現と構造学習、を行う。そして、前述の目隠しをして移動された後では、オンラインの処理として、(c) 学習した BN に基づいて最適なセンシング行動を計画する。以下では、この順に処理の概要を述べる。

## 4 環境情報データの収集

### 4.1 中国人郵便配達員問題の解

本研究では、Fig.4 に示すようなオフィス環境を対象にシミュレーションを行なった。初めに、完全な環境情報を獲得するために、移動ロボットに環境内の各廊下を通過させる必要がある。本研究では、このために中国人郵便配達員問題 (Chinese postman problem)(Fig.1) を適用した。

移動ロボットのナビゲーションについては、基本的にはポテンシャル法を用いる。センサとしては、レーザーレンジセンサを1台搭載し、移動ロボット真正面の180方向の奥行き情報を獲得できるものと仮定する。各交差点で移動ロボットは、予め next node algorithm [2] により作られたコマンドリストから行動コマンドを得る。

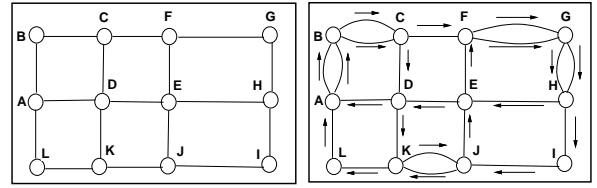


図 1: (left) A graph representing an office. (right) A path as a solution of Chinese postman problem.

Fig.4 の A, B, ..., L は交差点のラベルであり、移動する途中で人間が移動ロボットに対して教示する。実験環境では、廊下壁面の両側の所々に凹みがあるものとし、それらをランドマークとして用いる(注: ロボットに搭載するセンサが検出できるものであれば基本的には何でも良い)。このランドマークはロボット両側のレンジデータをフィルタリング処理して検出する。なお、本研究では、ランドマークは廊下(の一部)に対してユニークな存在ではなく、少数のランドマークの観測のみでは自己位置の決定に不確か性が残る場合を取り扱う。ただし、問題の簡単化のために、local distance<sup>1</sup>の不確か性については扱っていない。また、交差点の幾何的な特徴の識別には、サポートベクトルマシン (support vector machines)[6] を用いた。

### 4.2 環境情報データの記述

我々は隣交差点の間の情報は segment ( $S_g$ ) と定義している。一つの  $S_g$  は四つの記述を含んでいる。

1. 二つ交差点のラベル。
2. 二つ交差点間のランドマークの情報。
3. 移動ロボットは各交差点に入る途端にセンシングされた交差点の幾何的な特徴。
4. 移動ロボットは各交差点に入る時の行動。

以上のデータを、移動ロボットは移動しながら獲得し、これに基づいて、BN のパラメータと構造を学習する。

## 5 データから BN の構造学習

本研究では BN の構造学習をするためにスコアに基づくアルゴリズムに K2[3][8] を導入した。K2 は greedy search であるが、具体的には以下ようになる。始めに、各ノードの親ノードがない状態にしておく。そして、用意された親ノードを一つずつ加えてグラフを作る。この

<sup>1</sup>隣ランドマーク同士の間、又は交差点と隣のランドマークの間のローカル距離

グラフをベイジアンスコア [3] により評価し、増加するノードだけ親として残る。グラフのベイジアンスコアが最大になったとき、または、用意している親ノードがなくなると他のノードへ移る。すべてのノードは、以上の順序で親ノードを探索し、最適な構造を生成する。しかし、構造空間を全探索するのは膨大な計算になる。例えば、ノードの数  $n = 2$  だと可能な構造は 3 個、 $n = 3$  ならば可能な構造は 25 個、 $n = 5$  ならば 29,000 個の可能な構造がある。探索空間を減らすために、 $K2$  は親子順序で探索空間を制限するオーダリングがあるが、オーダリングを得ることは容易ではない。ここでは、文献 [4] が用いたように、遺伝アルゴリズム (GA)[9] を使って準最適なノードの順序づけを探索する。得られたノードの順序づけを  $K2$  に用いて準最適な  $BN$  構造を求めた。

### 5.1 $BN$ のノード設計

一例であるが、13 個確率変数を持つ  $BN$  があって、ノード  $Head(H)$ ,  $Mid(M)$ ,  $Tail(T)$  は二つ隣の廊下にある三つ交差点のラベルを表している<sup>2</sup>。交差点ラベルノードの確率変数の数は 12 個がある、 $A, B, \dots, L$  で表現する。ノード  $Action1(a_1)$  と  $Action2(a_2)$  は各廊下に入る時の移動ロボットの行動を定義し、確率変数は三つがあって、*go forward*, *turn left*, *turn right* である。ノード  $Hf$ ,  $Mf$ ,  $Tf$  は移動ロボットが各廊下に入る所でセンシングした交差点の幾何的な特徴である。例えば、 $\perp, +, \top$  など、6 個がある。ノード  $m_{h1}, m_{h2}, m_{t1}, m_{t2}$  は廊下内のランドマークを示す。ノードには *local distance* の情報も含まれている。ランドマークは、(幾何的な特徴、*local distance*) というベクトルの形で表現する。このノード(確率変数)の数は 4 個である。データベースにある各データケースのラベルを *mediating variable*[1] として定義する ( $Cn$ )、ノード(確率変数)の数は 138 である。

### 5.2 $BN$ の構造学習

GA 母集団のサイズは 80 である、遺伝子は交差と突然変異の操作を行なって、100 世代探索した結果を Fig.2 に示す。点線は途中の個体群の平均 fitness スコアを示し、連続線は各世代のベスト fitness スコアを示す。100 世代の探索により、準最適なノードの順序づけを獲得する。Fig.3 で示すように、このノード順序づけを用いて準最適な  $BN$  の構造を生成する。

<sup>2</sup> $Head(H)$  は第一番目の交差点である、 $Mid(M)$  は二つ廊下間の交差点である。 $Tail(T)$  は最後の交差点である

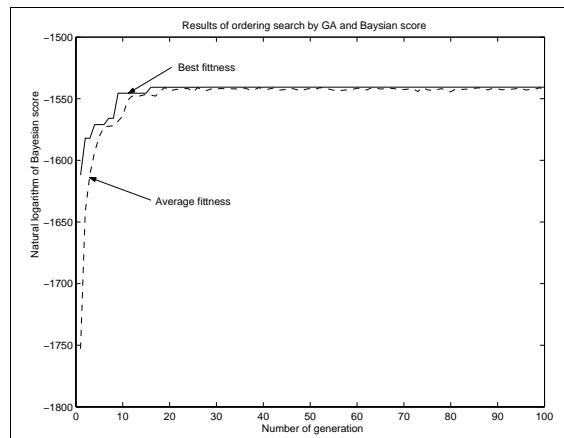


図 2: The results of ordering searching by GA and Bayesian score.

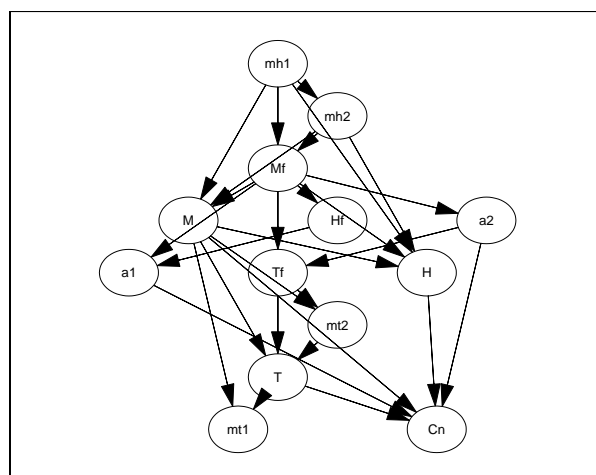


図 3: Learned  $BN$ 's structure by  $K2$  and  $GA$ .

## 6 センサプランニング

位置決め処理を単純化にするため、本研究では以下のような仮定を設けた：

1. 廊下内のランドマークは人間は設計した。システムは環境内どちらのランドマークを注目すべきかについては知らない。(環境から  $BN$  用のランドマークを学習する問題は今後の研究課題である。)
2. センサ情報は全部離散値である、もしも移動ロボットが得たセンサ情報が連続値であれば、それを離散値化して  $BN$  の推論アルゴリズムにフィットさせなければならない。

本センサプランニングシステムは三つのステップから成る：(1) 位置決めのための推論、(2) センシング行動のための予測、(3) 位置決めのためのセンシング行動計画。

**Input:**

- a. *BN engine*: **engine**,
- b. possible action list: **actlist**,
- c. 2-D list of predicted sensing information according to **actlist**: **Psel**
- d. obtained sensing information list: **Osel**
- g. list: **ActSenNum**, the number of predicted sensing information sets.  
For example, if **ActSenNum** is (4, 5), it means there are 4 and 5 predicted sensing information sets which according to the first and the second possible action of **actlist**, respectively. The list's length is length of list **actlist**.

**Output:**

- a. optimal action **OptAction**
  - b. action-sensing list **ActSenlist**, each member of list **ActSenlist** is a vector which include an element of **actlist**, and a refined sensing list corresponding to the element.
- var:**
- a. Evidence list, **SenEven** assigns a part of **Psel**.
  - b. Global localization belief **TolBef**.
  - c. the number of classifiable the robot's location: **count**.
  - d. sensing cost: **SenCost**.
  - e. double: **Cost**.

**Sensor Planning:**

```

Begin
  for  $i = 1 : \text{length}(\text{actlist})$ 
    count = 0;
    SenCost = 0;
  A   for  $k = 1 : \text{ActSenNum}(i)$ 
        for  $j = 1 : \text{length}(\text{Psel}(i))$ 
          push Psel(i) into SenEven;
        * [ActSenlist] =
          BayesianNetworkInference(engine,
            actlist(i), Osel, SenEven);
          Calculate TolBef3;
          if (TolBef  $\geq 2.7^4$ ),
  B   count ++;
      Cost = CaculateSensingCost(SenEven)5
      clear SenEven's content;
      if (SenCost < Cost),
        SenCost += Cost;
      end if
    end if
    goto: A;
  end for
  end for
  ActSenlist( $i, 1$ ) = actlist( $i$ );
  ActSenlist( $i, 2$ ) = count;
  ActSenlist( $i, 3$ ) = SenCost;
  end for
  goto: C; (Table 2)
end

```

表 1: Sensor Planning Algorithm (Definition of variables)

位置決めのための推論：移動ロボットは作業空間から得られたセンサ情報を *BN* に入力し、自分にいる廊下の交差点ラベルノードの確率はグローバル位置決め信念度の評価になる。もしも、二つノードの確率が 0.9 以上になれば、位置決めプロセスは完了する。そうでなければ、移動ロボットはもっと必要な情報を得るために、位置の不確実な交差点でセンシング行動のプランを立て始める。

センシング行動のための予測は二つステップがある。STEP ONE、得られたセンサ情報に基づいてノード  $C_n$  の非零な変数値を探し、一つのリストに保存する。STEP TWO、非零な  $C_n$  の変数値と得られたセンサ情報に基づいて可能な行動とこれらの行動に対応するセンサ情報を予測し、一つの行列に保存する。

位置決めのためのセンシング行動計画は三つのステップから成る。全体の詳細な流れを Table 1 と上の手順で示す。STEP ONE、得られた情報、可能な行動とこれらの行動に対応するセンサ情報に基づいて、*global localization belief*(**TolBef**)(the line  $\star$ ) を推論する。STEP TWO、センシングコストとロボットは識別できる *localization* の数 (**count**)<sup>6</sup> を計算する。STEP THREE、最大 **count**、又は、相対的に小さいセンシングコストを持つ最適な行動を見つける。Table 2 に STEP THREE の詳細な手順を示す。

<sup>3</sup>adding of maximal probability of nodes  $H, M, T$ .

<sup>4</sup>We define if the probability of node *Head*, *Mid* and *Tail* is not less than 9.0, the mobile robot can ensure its location, and  $0.9 * 3 = 2.7$ .

<sup>5</sup>The sensing cost is calculated by adding the local distance of **SenEven**'s element

<sup>6</sup>もし、移動ロボットは推論した三つ交差点の最大確率が 0.9 より大きい、つまり、それらの総和が 2.7 より大きければ、システムはロボットがこの三つ交差点にある廊下にいることを決定できる、三つ交差点にある廊下はロボットの location である。

```

C      Function:
OptAction = GetOptimalAction
(ActSenlist)
Begin
  sort the list ActSenlist in the
  2th column's elements of ActSenlist
  that the have been stored line B;
  OptAction = ActSenlist(1, 1);
  for x = 2 : length(actlist)
  if ((ActSenlist(x, 2) ≥ ActSenlist(1, 2)) &
      (ActSenlist(x, 3) < ActSenlist(1, 3))),
      OptAction = ActSenlist(x, 1);
  end if
  end for
end(GetOptimalAction)

```

表 2:

## 7 実験

実験には、MATLABの Bayes Net Tool Box[8] 及び、GA Tool Box[9] を用いた。

処理の一例を以下に示す。移動ロボットは交差点  $D$  から出発し(もちろん、ロボットは自分は  $D$  にいることが分らない)、廊下両側のセンサ情報をセンシングしながら、 $BN$  より *global localization* 信念度を推論する、つまり、自分がいる廊下の交差点のラベル、 $Head(H)$ 、 $Mid(M)$  と  $Tail(T)$  の確率を推論する。

移動ロボットが交差点  $K$  に到着した時の  $Head(H)$ 、 $Mid(M)$  と  $Tail(T)$  の確率は Fig.4 のようになる。Fig.4 で、括弧つきの実数、黒い背景と括弧つきの実数、又は陰影と括弧つきの実数で表した値はそれぞれノード  $Tail(T)$ 、 $Mid(M)$ 、と  $Head(H)$  確率に対応している。ノード  $Tail(T)$ 、 $Mid(M)$ 、と  $Head(H)$  の確率変数の値がある交差点のラベルである確率をその交差点の場所に表示している。Fig.4(a) に示すように各交差点の  $Head(H)$ 、 $Mid(M)$ 、 $Tail(T)$  の信念度の値は小さいので、移動ロボットは交差点  $K$  まで移動するだけでは自己位置が決定できない。従って、ロボットは交差点  $K$  でセンシング行動の計画を行う。システムが予想した行動は *turn left* と *turn right* である。この二つの行動とこれらに対応するセンサ情報を基づいて、*global localization* 信念度とセンシングコストを評価する。もしも移動ロボットが *turns left* を取ると、 $Tail$  交差点  $J$  まで行かなければ十分なセンサ情報を得られない。また、もしも移動ロボットが *turn right* を取ると、Fig.4(b) 示すよう

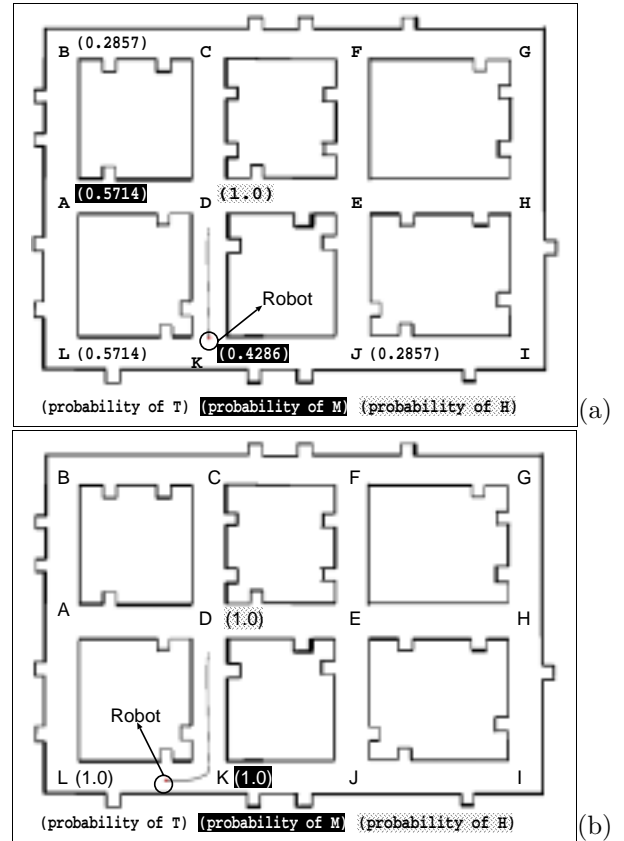


図 4: An example of the experiments of sensor planning for the mobile robot localization. (In the figure, the real numbers in ( ), ( ) with black square, ( ) with hatched square represent probability of node T, M, H, respectively. If the intersection is the instantiation of node T, M, H, the probability is shown at the intersection.)

にロボットはより低いコストで十分なセンサ情報を得られる。従って、自己の位置決めのための最適な行動は *turn right* である。

## 8 結論

本論文では、不確実性を含むロボット位置決め問題を解決するためのセンサプランニング手法を提案した。システムは GA を結合した K2 アルゴリズムを用いてベイジアンネットワークの構造を環境データから学習する。 $BN$  は得られたセンシング情報により、可能なセンサ行動を予測し、又は、それらの行動に対応するセンシング情報も連想する。さらに、グローバルな位置決め信念度とセンシングコストのバランスを考慮して最適な行動を決める。オフィス環境を想定したシミュレーション実験により、本システムの有効性を確かめた。

今後の課題は、実データを使って学習とセンサプラン

ングのアルゴリズムの有効性を検証することである。

## 参考文献

- [1] F. V. Jensen, "Bayesian networks and Decision Graphs," *Springer*, 2001.
- [2] J. Edmonds and E. L. Johnson, "Matching Euler Tours and the Chinese Postman," *Mathematical Programming*, 5, pp.88-124, 1973.
- [3] G. Cooper and E. Herskovits, "A Bayesian method for the induction of probabilistic networks from data," *Machine Learning*, 9:309-347, 1992.
- [4] P. Larranaga, C. Kuijpers, R. Murga, Y. Yurramendi, "Learning Bayesian network structures by searching for the best ordering with genetic algorithms," *IEEE Transactions on System, Man and Cybernetics*. Vol 26. No. 4, 487-493, 1996.
- [5] D. Corney. "Designing Food with Bayesian Belief Networks". *Fourth International Conference on Adaptive Computing in Design and Manufacture (ACDM2000)*, April 26th - 28th, University of Plymouth, UK.
- [6] N. Cristianini and J. Shawe-Taylor, "An Introduction to Support Vector Machines," *Cambridge University Press*, 2000.
- [7] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. "Robust Monte Carlo Localization for Mobile Robots", *Artificial Intelligence (AI)*, 2001.
- [8] Kevin Murphy. "The Bayes Net Toolbox for Matlab", *Computing Science and Statistics*, vol 33, 2001. Software available at <http://www.cs.berkeley.edu/~murphyk/Bayes/bnt.html>.
- [9] Chris Houck, Jeff Joines, and Mike Kay. "A Genetic Algorithm for Function Optimization: A Matlab Implementation". Technical Report NCSU-IE 95-09, 1995. Software available at <http://www.ie.ncsu.edu/mirage/GAToolBox/gaot/>.
- [10] H. Asoh, Y. Motomura, I. Hara, S. Akaho, S. Hayamizu, and T. Matsui, "Combining Probabilistic Map and Dialog for Robust Life-long Office Navigation," *Proc. of the Int. Conf. on Intelligent Robots and Systems (IROS'96)*, pp.880-885, 1996.
- [11] J. Miura and Y. Shirai, "Vision-Motion Planning for a Mobile Robot considering Vision Uncertainty and Planning Cost," *Proc. 15th Int. Joint Conf. on Artificial Intelligence*, pp.1194-1200, 1997.
- [12] R. Rimey and C. Brown, "Control of Selective Perception using Bayes Nets and Decision Theory," *Int. Journal of Computer Vision*, Vol.12, pp.173-207, 1994.
- [13] Kenneth Basye, Thomas Dean, Jak Kirman, and Moises Lejter, "A Decision-Theoretic Approach to Planning, Perception, and Control", *IEEE Expert*, Vol.7, Number 4, Pages 58-65, 1992.
- [14] T. Dean et al., *Artificial Intelligence*, The Benjamin/Cummings, 1995.
- [15] D. Heckerman. "A Bayesian approach to learning causal networks". *Technical Report MSR-TR-95-04, Microsoft Research*, March, 1995.
- [16] S. Kristensen, "Sensor Planning with Bayesian Decision Analysis", *PhD thesis*, Faculty of Technology and Science, Aalborg University, Aalborg, Denmark, 1996.