

Zastosowanie neuronowych sieci komórkowych w nawigacji

Streszczenie:

W pracy zaprezentowano fragmenty systemu nawigacyjnego robota ruchomego wyposażonego w laserowy czujnik odległości. Przyjęto rastrowy model sceny. Przedstawiono zastosowanie sieci komórkowych w procesie planowania ścieżki i systemie samolokalizacji robota

Abstract:

A navigation system for an autonomous mobile robot is described in the paper. The scene is represented as a grid of cells. Cellular neural network is used for the path planning and robot's localization

1 Wstęp

Jednym z najważniejszych problemów w dziedzinie robotyki jest konstruowanie systemu planowania działań ruchomego robota, w szczególności planowania optymalnej i bezkolizyjnej trajektorii. System taki powinien umożliwić bezpieczne przeprowadzenie robota od zadanej pozycji do celu. Jeżeli przyjmiemy, że przeszkody oraz cel mogą się poruszać oraz przedmioty znajdujące się w otoczeniu robota mogą mieć dowolne kształty, to problem planowania ścieżki nie jest zadaniem trywialnym. Szczególnie trudnym zadaniem jest planowanie ścieżki w nieznanym otoczeniu. W poniższym opracowaniu zaproponowano system umożliwiający bezpieczne poruszanie się robota w nieznanym otoczeniu. Zakładamy, że znamy początkowe położenie robota i celu. Przedstawiony system składa się z następujących modułów: tworzenie i aktualizacja mapy otoczenia, planowanie na podstawie utworzonej mapy bezkolizyjnej trajektorii, określenie położenia robota w globalnym układzie współrzędnych. Robot wyposażony jest w laserowy czujnik odległości. Na podstawie wskazań czujnika tworzona jest rastrowa mapa otoczenia. Sposób tworzenia i aktualizacji mapy przedstawiono w [4]. Aby robot mógł tworzyć globalną mapę otoczenia niezbędna jest znajomość jego położenia w globalnym układzie współrzędnych. Położenie to jest określane na podstawie znajomości położenia obiektów charakterystycznych - ścian. W trakcie prowadzonych w ramach grantu KBN pt. "Badania metod i zastosowanie techniki laserowej do wspomaganie nawigacji robotów mobilnych" badań testowano wiele metod planowania bezkolizyjnej ścieżki. Najbardziej efektywną okazała się metoda wykorzystująca sieć komórkową. W poniższym opracowaniu przedstawione zostaną podstawowe własności neuronowych sieci komórkowych i ich zastosowanie w planowaniu ścieżki i lokalizacji robota.

2 Sieci komórkowe

Sieci komórkowe są 2,3 lub n-wymiarowymi tablicami zawierającymi elementarne elementy dynamiczne nazywane komórkami lub neuronami, które oddziałują na siebie jedynie lokalnie[1]. Komórki sieci połączone są jedynie z komórkami należącymi do ich r-sąsiedztwa.

R-sąsiedztwem N_r^i neuronu C_{ij} jest zbiór neuronów spełniających warunek:

$$N_r^{ij} = \{C_{kl} : \max(|k-i|, |l-j|) \leq r\} \quad (1)$$

Schemat połączeń jest identyczny dla wszystkich, poza komórkami brzegowymi elementów sieci. Sygnałami sterującymi komórką C_{ij} są: sygnały wyjściowe y_{kl} neuronów należących do N_r^{ij} , sygnały wejściowe u_{kl} neuronów należących do N_r^{ij} oraz sygnał I zwany polaryzacją. Współczynniki wagowe sygnałów wyjściowych komórek należących do N_r^{ij} nazywane są współczynnikami sprzężenia zwrotnego i oznaczone są symbolem a_{ij}^{kl} . Współczynniki wagowe sygnałów wejściowych komórek należących do N_r^{ij} nazywane są współczynnikami sterowania i oznaczone są przez b_{ij}^{kl} . Współczynniki wagowe określają stopień oddziaływania komórek należących do N_r^{ij} na neuron C_{ij} . Symbolem x_{ij} oznaczony jest stan komórki C_{ij} .

Dynamikę zmian w sieci komórkowej działającej w sposób dyskretny opisuje równanie:

$$x_{ij}(t+1) = \sum_{k=-r}^r \sum_{l=-r}^r a_{ij}^{i+k, j+l} y_{i+k, j+l}(t) + \sum_{k=-r}^r \sum_{l=-r}^r b_{ij}^{i+k, j+l} u_{i+k, j+l}(t) \quad (2)$$

$$y_{ij}(t+1) = f(x_{ij}(t)) \quad (3)$$

Ponieważ schemat połączeń jest identyczny dla wszystkich neuronów, więc dla dowolnych dwóch komórek C_{ij} i C_{kl} , wagi połączeń w obu kierunkach są identyczne tzn. zachodzi warunek:

$$a_{i+n, j+m}^{ij} = a_{k+n, j+m}^{kl} \quad b_{i+n, j+m}^{ij} = b_{k+n, j+m}^{kl} \quad (4)$$

Współczynniki a_{ij}^{kl} i b_{ij}^{kl} zapamiętywane są w postaci macierzy A i B o wymiarach $(2r+1) \times (2r+1)$. Zbiór $\{A, B, f\}$ nosi nazwę szablonu sieci (ang. cloning template).

Wynikiem przetwarzania sygnałów w sieci jest zbiór wartości sygnałów wyjściowych w chwili osiągnięcia przez sieć stanu równowagi stabilnej. Równowaga stabilna zostaje osiągnięta, jeśli spełniony jest następujący warunek:

$$\lim_{t \rightarrow \infty} x_{ij}(t) = \text{const} \quad (5)$$

Stabilność sieci komórkowej zależy od funkcji f i postaci macierzy A i B .

3 Określanie położenia robota

Tworzenie globalnej mapy otoczenia jest możliwe jedynie wtedy, gdy znane jest dokładnie położenie robota w globalnym układzie współrzędnych. Większość obecnie stosowanych robotów posiada wbudowany sprzętowy system zliczeniowego określania położenia. Niestety informacja taka nie jest wystarczająca, jeśli robot realizuje bardzo długie ścieżki z dużą ilością manewrów typu gwałtowny skręt lub duża zmiana prędkości. Nowoczesny system powinien być wyposażony dodatkowy moduł, który na podstawie znajomości mapy globalnej otoczenia i położenia obserwowanych przez sensory przeszkód, koryguje dane dotyczące położenia robota. W systemach tych stosuje się najczęściej mechanizm filtru Kalmana. Zaczerpinęty z [5] algorytm działania modułu samolokalizacji robota jest następujący:

Założmy, że znamy położenie robota w chwili k , opisane wektorem:

$$x(k|k) = [x(k), y(k), \phi(k)]^T \quad (6)$$

oraz przesunięcie $T(k)$ i obrót $\phi(k)$, które robot powinien wykonać w sytuacji idealnej opisane wektorem:

$$u(k) = [T(k), \phi(k)] \quad (7)$$

Położenie robota w chwili $k+1$ można opisać równaniem:

$$x(k+1|k+1) = f(x(k), u(k)) + v(k) \quad (8)$$

gdzie $v(k)$ - jest szumem powstałym w wyniku niedokładności systemu sterowania.

$$f(x(k), u(k)) = \begin{bmatrix} x(k) + T(k) \cdot \cos(\phi(k)) \\ y(k) + T(k) \cdot \sin(\phi(k)) \\ \phi(k) + \delta\phi(k) \end{bmatrix} \quad (9)$$

Należy określić wielkość szumu $v(k)$. W chwili początkowej, na podstawie analizy wskazań lasera, robot ocenia położenie punktów charakterystycznych - ścian.

W chwili $k+1$ na podstawie znajomości położenia punktów charakterystycznych w chwili k i wektora $u(k)$ oblicza się odległość $z'_j(k+1)$ od tych obiektów charakterystycznych zgodnie ze wzorem:

$$z'_j(k+1) = h_{jt}(x(k+1|k), p_t) + w_j(k) \quad (10)$$

$$x(k+1|k) = [x(k) + T(k) \cdot \cos(\phi(k)), y(k) + T(k) \cdot \sin(\phi(k)), \phi(k) + \delta\phi(k)] \quad (11)$$

gdzie: $w_j(k)$ - jest szumem powstałym w wyniku niedokładnych wskazań sensorów, a p_t jest wektorem opisującym obiekt charakterystyczny.

Dla ściany $p_t = (x_1, y_1, x_2, y_2)$, gdzie (x_1, y_1) - początek odcinka, a (x_2, y_2) - koniec odcinka.

Następnie obliczana jest wielkość $\nu_{ij}(k+1)$

$$\nu_{ij}(k+1) = [z_j(k+1) - z'_i(k+1)] \quad (12)$$

i macierz S_{ij}

$$S_{ij} = \nabla h_i P(k+1|k) \nabla h_i^T + R_i(k+1). \quad (13)$$

$P(k+1|k)$ - jest macierzą wariancji, określającą niepewność położenia robota w chwili $k+1$, uwzględniając informacje zebrane w k poprzednich krokach. Macierz $R_i(k+1)$ określa niepewność położenia obserwowanych obiektów charakterystycznych.

W następnym kroku obliczana jest macierz W :

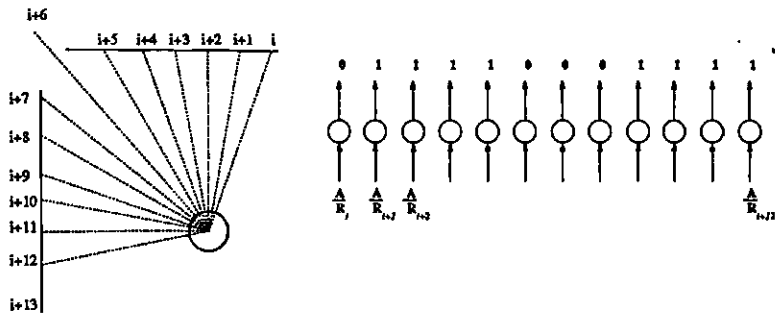
$$W(k+1) = P(k+1|k) \nabla h^T S^{-1}(k+1) \quad (14)$$

Nową poprawioną pozycję robota określa wzór:

$$x(k+1|k+1) = x(k+1|k) + W(k+1) \cdot \nu(k+1) \quad (15)$$

oraz związana z nowym położeniem wariancja określona jest wzorem:

$$P(k+1|k+1) = P(k+1|k) - W(k+1) \cdot S(k+1) \cdot W^T(k+1) \quad (16)$$



Rysunek 1: Wartości wyjściowe sieci dla zadanej sceny

Ze względu na dużą szybkość działania lasera głównym zadaniem jest znalezienie efektywnego algorytmu wykrywania obiektów charakterystycznych. W poniższym opracowaniu proponuje się zastosowanie jednowymiarowej sieci komórkowej do wykrywania fragmentów ścian. Odczyty lasera zapamiętywane są w postaci tablicy, której i -ty element przechwytuje odległość do przeszkody przy kącie skanowania $\phi_{rob} + i \cdot \delta\alpha$, gdzie ϕ_{rob} jest orientacją robota w globalnym układzie współrzędnych, a $\delta\alpha$ jest jednostkowym przyrostem kąta skanowania.

Sygnał wejściowy do sieci $u_i = \frac{A}{R_i}$, gdzie A jest pewną stałą, a R_i jest odległością wskazywaną przez laserowy czujnik odległości przy kącie skanowania $\phi_{rob} + i \cdot \delta\alpha$. Przyjęto sąsiedztwo pierwszego stopnia. Macierz współczynników sprzężenia jest macierzą zerową, macierz sterowania B ma następującą postać:

$$B = [-1, 2c, -1] \quad c = \cos(\delta\alpha) \quad (17)$$

Wartość sygnału wyjściowego jest następująca:

$$y_i = \begin{cases} 1 & \text{gdy } |\sum b_j \cdot u_j| \leq \epsilon \\ 0 & \text{gdy } |\sum b_j \cdot u_j| > \epsilon \end{cases} \quad (18)$$

Na rysunku 1 pokazano wartości wyjściowe neuronowej sieci komórkowej dla zadanej sceny. Jeśli istnieje ciąg $\{i, i+1, \dots, i+j\}$ taki, że $u_{i+1}=1, u_{i+2}=1, \dots, u_{i+j-1}=1$ to ciąg odczytów $\{R_i, \dots, R_{i+j}\}$ wyznacza wspólny ciąg punktów leżących zawierających się w odcinku



Rysunek 2: Niepewność położenia robota dla różnych obiektów charakterystycznych

o początku w punkcie (x_1, y_1) i końcu w punkcie (x_2, y_2) , gdzie

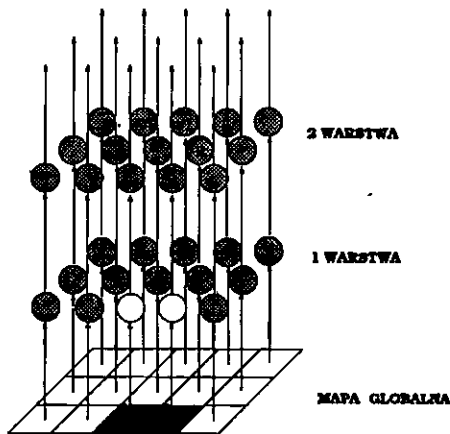
$$\begin{aligned} x_1 &= x_{rob} + R_i * \cos(\phi_{rob} + i \cdot \delta\alpha) \\ y_1 &= y_{rob} + R_i * \sin(\phi_{rob} + i \cdot \delta\alpha) \\ x_2 &= x_{rob} + R_{i+j} * \cos(\phi_{rob} + (i + j) \cdot \delta\alpha) \\ y_2 &= y_{rob} + R_{i+j} * \sin(\phi_{rob} + (i + j) \cdot \delta\alpha) \end{aligned} \quad (19)$$

Na rysunku 2 pokazano przykładową scenę. Kolem zaznaczono położenie robota, a elipsą niepewność jego położenia określoną przy pomocy filtru Kalmana. Strzałką zaznaczono kierunek w którym porusza się robot.

Jeśli położenie robota jest korygowane względem tylko jednego punktu charakterystycznego (ściany) to niepewność jego położenia jest większa niż w przypadku uwzględnienia kilku punktów charakterystycznych. Stopień niepewności zależy nie tylko od ilości punktów charakterystycznych, ale także od ich położenia względem robota.

4 Planowanie bezkolizyjnej ścieżki

Globalna metoda planowania ścieżki, proponowana w poniższej pracy, jest neuronalną realizacją i modyfikacją metody zaproponowanej przez Steels'a [3]. Architektura sieci przedstawiona jest na rysunku 3. Sieć składa się z dwóch warstw, z których każda jest siecią komórkową.



Rysunek 3: Architektura dyfuzyjnej sieci neuronowej

Każdej klatce mapy globalnej odpowiada para neuronów - jeden w pierwszej i jeden w drugiej warstwie. W sieciach komórkowych pierwszej i drugiej warstwy przyjęto sąsiedztwo stopnia jeden. Jeśli klatki R_{ij} i R_{kl} się stykają, to odpowiadające im neurony pierwszej warstwy C_{ij}^1 i C_{kl}^1 oraz neurony C_{ij}^2 i C_{kl}^2 drugiej warstwy są ze sobą w relacji r-sąsiedztwa. Sąsiedztwo to może być dowolnie rozszerzane, dzięki czemu wygenerowana ścieżka jest bardziej gładka i optymalna, ale w przypadku symulacji sieci neuronowej na komputerze działającym sekwencyjnie algorytm staje się bardziej skomplikowany obliczeniowo.

4.1 Warstwa wejściowa

Każda z klatek mapy wysyła sygnał wyjściowy u_{ij} o wartości 0, jeśli odpowiednia klatka jest zajęta przez przeszkodę i wartości 1 w przeciwnym przypadku. Przyjęto także, że

wartość sygnału u_{ij} dla wszystkich komórek brzegowych jest równą zero. W systemie założono, że wymiary robota mogą być większe niż wymiar klatki. Aby wygenerować bezkolizyjną trajektorię należy wymiary przeszkód rozszerzyć o połowę wymiaru robota. Temu celowi służy wprowadzenie funkcji S , która określa stan neuronu C_{ij} ; nie tylko w zależności od wartości sygnału u_{ij} , ale także od wartości sygnałów pochodzących z klatek sąsiednich. Wymiar sąsiedztwa zależy od wymiaru robota w klatkach np. jeśli rzut robota na płaszczyznę XY zawiera się w kwadracie o wymiarach 3×3 klatki, to wymiar sąsiedztwa wynosi 1. Przyjmujemy, że wszystkie elementy macierzy B mają wartość 1. Funkcja S ma następującą postać:

$$S(x) = \begin{cases} 0 & \sum_{kl \in N_{ij}} b_{kl}^{ij} \cdot u_{kl} < n \\ 1 & \sum_{kl \in N_{ij}} b_{kl}^{ij} \cdot u_{kl} \geq n \end{cases} \quad (20)$$

gdzie n jest liczbą elementów macierzy B .

4.2 Opis pierwszej warstwy

Pierwsza warstwa jest siecią komórkową działającą w czasie dyskretnym. Postać macierzy A przedstawiona jest na rysunku 4. Wartości α i β są liczbami z przedziału $(0,1)$ i odzwierciedlają odległości między odpowiednimi klatkami. Przyjęto, że $\alpha, \beta < 1$ i $\alpha < \beta$.

β	α	β
α	1	α
β	α	β

Rysunek 4: Macierz sprzężeń zwrotnych

Działanie pierwszej warstwy składa się z następujących etapów:

1. *Inicjalizacja wartości stanów* - neuron reprezentujący cel jest wyróżniony w pierwszej warstwie, neuron ten jest źródłem pobudzenia. Początkowe wartości stanów neuronów pierwszej warstwy w chwili początkowej są następujące:

$$x_{ij}(0) = \begin{cases} F \gg 1 & \text{gdy klatka } ij \text{ reprezentuje cel} \\ 0 & \text{w przeciwnym przypadku} \end{cases} \quad (21)$$

2. *Propagacja sygnałów wewnątrz sieci* - neuron pobudzony w chwili czasowej t pobudza neurony sąsiednie. Zmiany stanów komórek sieci w kolejnych chwilach czasowych określone są równaniami:

$$x_{ij}(t+1) = S\left(\sum_{kl \in N_{ij}^*} (b_{ij}^{kl} \cdot u_{kl}(t))\right) \cdot \max_{kl \in N_{ij}^*} (a_{ij}^{kl} \cdot y_{kl}(t)) \quad (22)$$

$$y_{ij}(t+1) = x_{ij}(t) \quad (23)$$

Powyższe równania nie są klasycznymi równaniami dynamiki sieci komórkowej, ale proces poszukiwania maksymalnego sygnałów wejściowych występuje w wielu rodzajach sieci, np. ART.

Propagacja sygnałów w pierwszej warstwie zachodzi aż do momentu osiągnięcia przez sieć stanu równowagi, tzn. aż do chwili \bar{t} takiej, że:

$$\forall_{i \geq \bar{t}} |x_{ij}(t) - x_{ij}(t+1)| \leq \epsilon \quad (24)$$

4.3 Generowanie ścieżki

Struktura drugiej warstwy jest identyczna jak pierwszej warstwy. Wszystkie wagi połączeń między neuronami mają wartość jeden. Sygnał pochodzący z warstwy pierwszej jest sygnałem wejściowym do warstwy drugiej i automatycznie uaktywnia odpowiedni neuron. W warstwie drugiej wyróżniony jest neuron odpowiadający pozycji robota. Jeśli neuron ten zostanie uaktywniony, to następuje sekwencyjny proces generowania ścieżki:

- Neuron reprezentujący pozycję robota jest wybierany jako fragment ścieżki.
- Jeśli neuron C_{ij}^2 reprezentuje fragment generowanej drogi, to kolejna pozycja robota wyznaczona jest przez neuron C_{kl}^2 , który spełnia następujący warunek:

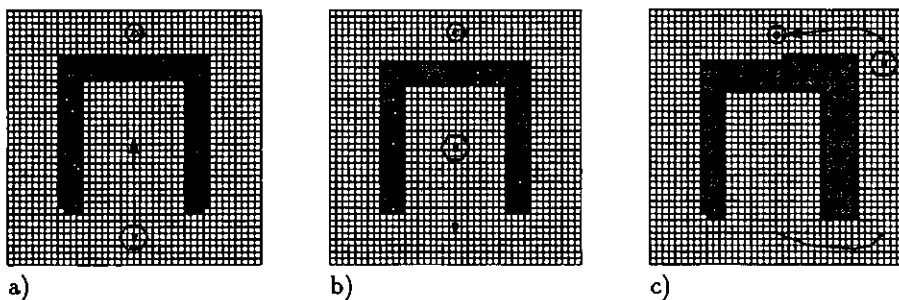
$$x_{kl}^2 = \max_{k \in N_i^j} x_{ij}^2 \quad (25)$$

- Proces generowania ścieżki jest zakończony, gdy kolejna pozycja robota wyznaczona będzie przez neuron spełniający warunek:

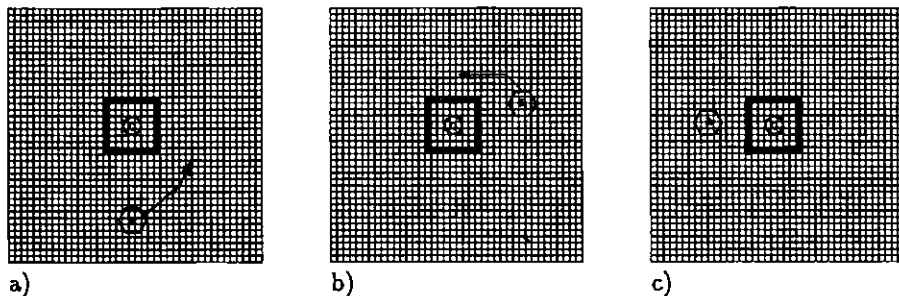
$$x_{kl}^2 = \max_{ij \in N^k} x_{kl}^2 \quad (26)$$

tzn. w sąsiedztwie neuronu C_{kl}^2 nie istnieje neuron o większym od niego pobudzeniu. Neuron ten reprezentuje pozycję celu.

Ciąg klatek odpowiadających wybranym neuronom drugiej warstwy wyznacza bezkolejną ścieżkę.



Rysunek 5: Planowanie ścieżki w przypadku występowania ślepej uliczki



Rysunek 6: Planowanie ścieżki, gdy cel otoczony jest przez przeszkody

Na rysunkach 5 - 6 przedstawiono wyniki symulacji komputerowej modułu planowania ścieżki i tworzenia mapy otoczenia. Założono, że wymiar klatki wynosi 10 cm, a robot

zajmuje obszar o wymiarach 3×3 klatki. Przedstawiono wyniki planowania w dwóch sytuacjach, w których klasyczne metody są nieskuteczne: występowania ślepej uliczki, lub gdy cel otoczony jest przez przeszkody. Szare kwadraty reprezentują rzeczywiste przeszkody, czarne kwadraty reprezentują klatki uznane za zajęte przez przeszkody, a białe klatki są wolne lub nieznanne. Mapa jest uaktualniana co 10cm (jedna klatka). Cel oznaczono literą G , a położenie robota literą R .

Na rysunkach 5a-5c przedstawiona sytuację, ślepej uliczki, wyraźnie widać, że po rozpoznaniu przeszkody robot automatycznie się wycofuje i planuje nową - prawidłową drogę do celu.

Opisana metoda dyfuzyjna ma następujące zalety:

- Neuronalna realizacja metody umożliwia wielorównoległe przeszukiwanie przestrzeni rozwiązań.
- Jeśli zostanie stworzony moduł umożliwiający w sposób ciągły odbieranie i przetwarzanie sygnałów z sensorów, to algorytm może być stosowany jako metoda reaktywnego sterowania.
- Umożliwia automatyczne przeplanowanie ścieżki w przypadku ślepej uliczki.
- Nie istnieją problemy związane z występowaniem minimów lokalnych.
- Umożliwia automatyczne wykrywanie sytuacji, w której cel lub robot otoczony jest przez przeszkody i wolna droga nie istnieje.

Cytowana literatura

- [1] Chua O., Young L.: "Cellular Neural Network Theory", *IEEE Transaction on Circuits and Systems* 35, str. 1257-1272, October 1988.
- [2] Kacprzyk T., Ślot K.: "Sieci neuronowe komórkowe", *Wydawnictwo Naukowe PWN*, Warszawa-Lódź, 1995.
- [3] Steels, L.: "Steps towards common sense", *Proceedings ECAI-88*, str. 49-54, Munchen 1988.
- [4] Racz J, Siemiątkowska B, Sawwa R., Petz M.: "2D Map Building Based on LRF Readouts", *Proceedings MMAR-96*, September, 1996, Międzyzdroje, Poland.
- [5] Leonard J.J., Durrant-Whyte H. F.: "Direct Sonar Sensing for Mobile Robot Navigation", *Kluwer Academic Publishers*, 1992.