Text zur Robotik2-Übung „Partikelfilter"

Dipl.-Ing. Florian Cordes

4. Januar 2011

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH

German Research Center for Artificial Intelligence
Research Group Robotics
Prof. Dr. Kirchner
Robert-Hooke-Str. 5
28359 Bremen, Germany

1 Generelles

1.1 Umweltinteraktion

Generell kann ein Roboter auf zwei Arten mit der Umwelt agieren:

- Messungen der Umweltparameter
  - Abstandssensoren, taktile Sensoren, Kameras
  - Die Messungen der Zeitpunkte zwischen t_1 und t_2 werden als $z_{t_1:t_2} = z_{t_1}, z_{t_1+1}, z_{t_1+2}, \ldots, z_{t_2}$ bezeichnet
  - Messdaten beinhalten Informationen über den gegenwärtigen Zustand der Umwelt des Roboters und des Roboters selbst. Diese Daten sind naturgemäß verrauscht.
- Aktionen durch Aktoren
  - Manipulation von Objekten, Roboterbewegung...
  - Auch Interaktion, wenn Roboter ,,einfach nur so da steht''
  - Kontroll-Aktionen zwischen t_1 und t_2 werden als $u_{t_1:t_2} = u_{t_1}, u_{t_1+1}, u_{t_1+2}, \ldots, u_{t_2}$ bezeichnet
  - Kontroll-Daten beinhalten Informationen über die Änderung des Zustands der Umwelt. (Setze Geschwindigkeit 10cm/s für 5s → der Zustand (Position) sollte um 50cm gewechselt haben → Fehler sind aber auch hier möglich!)

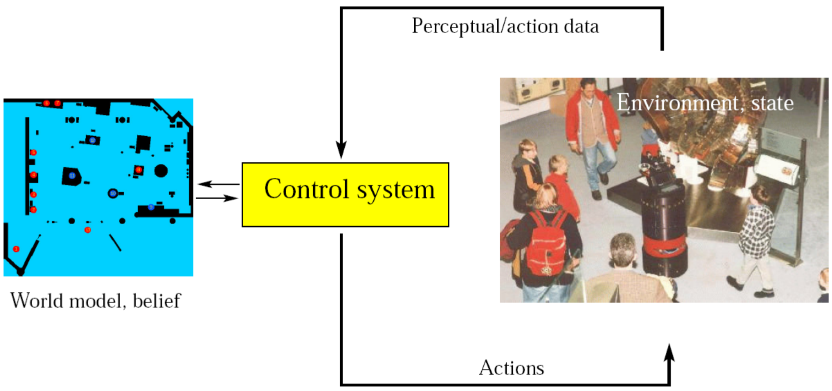

Abbildung 1.1: Robot environment interaction

Umweltwahrnehmung gibt Informationen über die Umgebung des Roboters, während Steuerbefehle, vor allem Bewegung, einen Verlust an Informationen über die Umgebung bzw. den Gesamtzustand bedeutet.

1.2 Vollständiger Zustand, Unvollständiger Zustand

Was kann der Zustand eines Roboters sein? Was fällt hier mit rein?

Wenn man einen absolut vollständigen Zustand eines Roboters wissen wollte, müsste man alles mit einbeziehen:

- Alle Aspekte der Umgebung, die einen Einfluss auf die Zukunft haben könnten,

- den Roboter selber, den Inhalt seines Speichers,
- das Wissen der Leute in der Umgebung.

Dies ist natürlich nicht möglich, daher muss man sich auf die Parameter beschränken, die ausreichen, um eine zuverlässige Aussage über den Zustand des Roboters und die unmittelbare Zukunft zu machen.

Der Zustand wird als vollständig bezeichnet, wenn das Wissen über vergangene Zustände, Messungen oder Kontroll-Befehle keine weitere Genauigkeit zu der Vorhersage der Zukunft beiträgt.

1.3 Dynamic Bayes Network

Genau genommen hängt der Zustand eines Roboters von allen vorherigen Zuständen, Messungen und Control-Inputs ab. Dies ergibt dann eine Wahrscheinlichkeit für einen Zustand x_t:

$$p(x_t|x_{0:t-1}, z_{1:t-1}, u_{1:t}) \tag{1.1}$$

Da der Zustand x_{t-1} bereits aus den davorhergehenden Zuständen, Messungen und Control-Inputs besteht, reduziert sich die Abhängigkeit zu

$$p(x_t|x_{0:t-1}, z_{1:t-1}, u_{1:t}) = p(x_t|x_{t-1}, u_t). \tag{1.2}$$

Die Wahrscheinlichkeit für den Zustand x_t ergibt sich also rekursiv aus dem vorherigen Zustand und dem letzten Control-Input. Diese Verteilungsfunktion wird Zustandsübergangswahrscheinlichkeit (State Transition Probability, PR S.25) bezeichnet und beschreibt, wie der Zustand des Roboters über die Zeit als Funktion der Kontroll-Kommandos u_t verändert wird.

Eine Messung z_t kann über eine Wahrscheinlichkeitsfunktion "vorausgesagt"werden, wenn der Zustand des Roboters vollständig ist, so ist diese Messung nur von dem derzeitigen Zustand abhängig:

$$p(z_t|x_{0:t}, z_{1:t-1}, u_{1:t}) = p(z_t|x_t). \tag{1.3}$$

Diese Verteilung ist die Messwahrscheinlichkeit (Measurement Probability) und beschreibt, mit welcher Wahrscheinlichkeit die Messungen z vom Zustand x erzeugt werden. Messungen können also als verrauschte Abbildungen des Zustands angesehen werden.

Abb. 1.2 zeigt den Ablauf von Zuständen und Messungen des Roboters. Der Zustand des Roboters am Zeitpunkt t ist stochastisch abhängig von dem zum Zeitpunkt $t-1$ und dem Control-Input u_t. Die Messung z_t wiederum hängt vom Zustand x_t ab. Man kann die Messung also als (verrauschte) Darstellung des Zustands sehen. Der Zustand –Wahrscheinlichkeitsverteilung des Zustands: $p(x_t|z_{1:t}, u_{1:t})$– selber ist nicht direkt zugänglich.

1.4 Belief-Funktion

Was ist eine Belief-Funktion?
Ein ,,Belief" spiegelt das interne Wissen über den Zustand der Umgebung wieder. Dies ist immer mit Unsicherheit behaftet, da der Zustand der Umgebung nicht vollständig gemessen werden kann. Allein die Position beispielsweise mit GPS zu bestimmen, ist schon nicht mit hundertprozentiger Genauigkeit möglich. Stattdessen muss der Roboter seinen Zustand / den

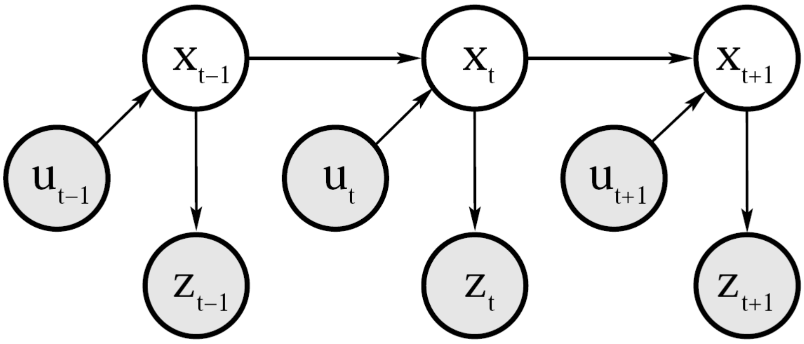

Abbildung 1.2: The dynamic Bayes network that characterizes the evolution of controls, states, and measurements.

Zustand der Umwelt anhand der ihm zur Verfügung stehenden Daten abschätzen. Daher wird zwischen dem tatsächlichem Zustand und dem Belief unterschieden.

Der Belief wird wird durch bedingte Wahrscheinlichkeiten (conditional probability) ausgedrückt. Eine Belief-Funktion ordnet jeder Hypothese über den Zustand eine Wahrscheinlichkeitsdichtefunktion zu. Belief-Funktionen sind a-posteriori-Verteilungen von Zustandsvariablen, d.h. eine Wahrscheinlichkeit für den Zustand, nachdem die Messungen und Control-Inputs berücksichtigt wurden:

$$bel(x_t) = p(x_t | z_{1:t}, u_{1:t}) \tag{1.4}$$

Es gibt auch eine Belief-funktion, die aufgestellt wird, bevor die letzte Messung stattfindet:

$$\overline{bel}(x_t) = p(x_t | z_{1:t-1}, u_{1:t}) \tag{1.5}$$

Dies ist die Vorhersage (prediction) des Zustandes, bevor die Messung stattgefunden hat. Die Berechnung von $bel(x_t)$ aus $\overline{bel}(x_t)$ wird Mess-Update bezeichnet.

—

Die A-posteriori-Verteilung gewichtet Wahrscheinlichkeitswerte eines Modells nach der Beobachtung von Ereignissen, die durch das Modell beschrieben werden. Mit Hilfe des Satzes von Bayes können diese auch aus der A-priori-Verteilung berechnet werden.

—

1.5 Mathematische Herleitung Bayes-Filter

Bayers Rule:

$$p(x|y,z) = \frac{p(y|x,z) \cdot p(x|z)}{p(y|z)} \qquad \text{(2.16), S.17 PR} \tag{1.6}$$

Man kann nun Bayers Regel auf die Wahrscheinlichkeitsverteilung des Zustands anwenden und erhält:

$$bel(x_t) = p(x_t|z_{1:t}, u_{1:t}) = \frac{p(z_t|x_t, z_{1:t-1}, u_{1:t}) \cdot p(x_t|z_{1:t-1}, u_{1:t})}{p(z_t|z_{1:t-1}, u_{1:t})} \tag{1.7}$$

$$= \eta \cdot p(z_t|x_t, z_{1:t-1}, u_{1:t}) \cdot p(x_t|z_{1:t-1}, u_{1:t}) \tag{1.8}$$

Wenn wir nun die Messung z_t voraussagen wollen, sind keine weiteren Informationen als der aktuelle Zustand notwendig, da die letzten Zustände keine weitere Information tragen. Daher können wir die folgende konditionale Unabhängigkeit ausnutzen:

$$p(z_t|x_t, z_{1:t-1}, u_{1:t}) = p(z_t|x_t) \tag{1.9}$$

Damit kann Ausdruck weiter vereinfacht werden:

$$p(x_t|z_{1:t}, u_{1:t}) = \eta \cdot p(z_t|x_t) \cdot p(x_t|z_{1:t-1}, u_{1:t}) \tag{1.10}$$

$$\Rightarrow bel(x_t) = \eta \cdot \underbrace{p(z_t|x_t)}_{\text{Measurement Probability}} \cdot \overline{bel}(x_t) \tag{1.11}$$

Unter Ausnutzung des Zusammenhangs der Totalen Wahrscheinlichkeit

$$p(x) = \int p(x|y) \cdot p(y) dy \tag{1.12}$$

können wir $\overline{bel}(x_t)$ erweitern zu:

$$\overline{bel}(x_t) = \int p(x_t|x_{t-1}, z_{1:t-1}, u_{1:t}) \cdot p(x_{t-1}|z_{1:t-1}, u_{1:t}) dx_{t-1} \tag{1.13}$$

Wenn wir x_{t-1} kennen, haben vergangene Messungen und Control-Inputs keine zusätzliche Information über x_t. Damit vereinfachen wir den ersten Multiplikator zu

$$p(x_t|x_{t-1}, z_{1:t-1}, u_{1:t}) = p(x_t|x_{t-1}, u_t) \tag{1.14}$$

Dabei müssen wir das u_t natürlich stehen lassen, da dieses nicht in x_{t-1} enthalten ist.

Im zweiten Multiplikator kann das u_t dagegen ohne Probleme raus gelassen werden, da dieses erst nach dem Zustand x_{t-1} auftritt. Damit ergibt sich die rekursive Berechnung der Belief-Funktion:

$$\overline{bel}(x_t) = \int \underbrace{p(x_t|x_{t-1}, u_t)}_{\text{State Transition Probability}} \cdot \underbrace{p(x_{t-1}|z_{1:t-1}, u_{1:t-1})}_{bel(x_{t-1})} dx_{t-1} \tag{1.15}$$

Damit haben wir dann den Grundsatz für Bayes-Filter hergeleitet: die Rekusive Berechnung. Wenn wir einen bekannten Ausgangszustand haben, können wir die darauf folgenden Zustände berechnen. D.h. wir nehmen eine zufällige Verteilung von Zustandsvariablen an, und verbessern die Abschätzung des Zustandes sukzessive in jedem Filterungsschritt durch das Einbinden weiterer Messungen. Dies ist die Grundlage auch für den Partikelfilter.

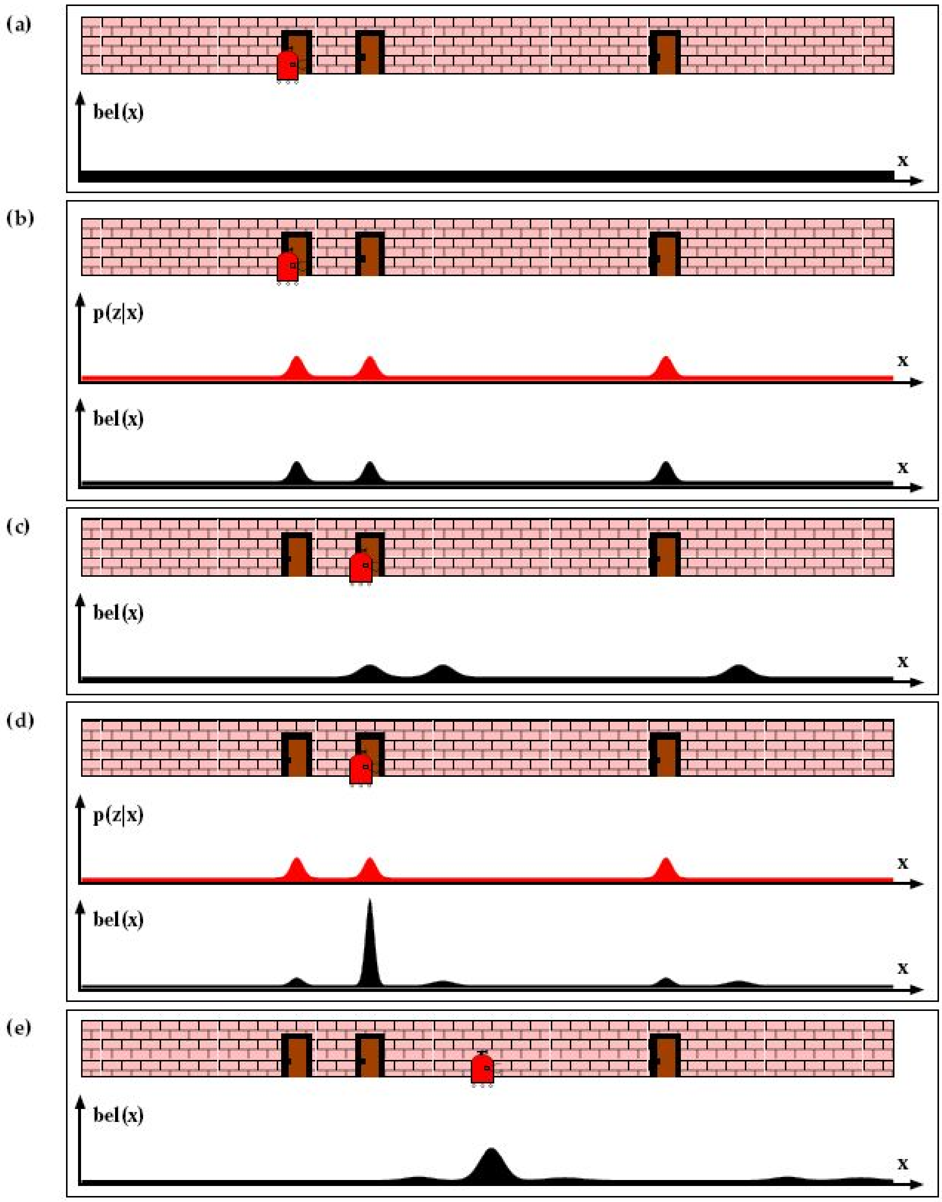

Abbildung 1.3: Update der Belief-Funktion

2 Partikel Filter

Der Partikelfilter ist eine spezielle Unterart in der Gruppe der Bayes-Filter. Partikelfilter werden eingesetzt, um unsichere Zustände abschätzen zu können. Was kann die sichere Wahrnehmung beeinflussen?

- Begrenzter Sichtbereich
- Begrenze Auflösung
- Rauschen der Sensoren
- Dynamische Umgebungen
- Zufällige Änderungen

Die Partikel des Partikelfilters sind diskrete Abschätzungen des aktuellen Zustands des Roboters. Die Partikel eines Partikelfilters werden mit

$$\mathcal{X}_t := x_t^{[1]},\ x_t^{[2]},\ \ldots,\ x_t^{[M]} \qquad M\text{: Anzahl der Partikel} \tag{2.1}$$

beschrieben.

Das Partikel-Set soll die Belief-Funktion $bel(x_t)$ approximieren. Je mehr Partikel in einer Region des Zustandraumes sind, desto wahrscheinlicher ist es, dass der aktuelle Zustand in diesem Bereich liegt.

Partikelfilter sind rekursiv, d.h., sie bestimmen das neue Set von Partikeln $\mathcal{X}_t$ aus dem vorhergehenden Set von Partikeln $\mathcal{X}_{t-1}$ zusammen mit dem letzen Steuerbefehl u_t und der letzten Messung z_t.

01: AlgorithmusPartikel-Filter($\mathcal{X}_{t-1}, u_t, z_t$):
02: $\overline{\mathcal{X}}_t = \mathcal{X}_t = \emptyset$
03: for m=1 to M do
04: sample $x_t^{[m]} \sim p(x_t | u_t, x_{t-1}^{[m]})$
05: $w_t^{[m]} = p(z_t | x_t^{[m]})$
06: $\overline{\mathcal{X}}_t = \overline{\mathcal{X}}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$
07: endfor
08: for m=1 to M do
09: draw i with probability $\propto w_t^{[i]}$
10: add $x_t^{[i]}$ to $\mathcal{X}_t$
11: endfor
12: return $\mathcal{X}_t$

1. Kopfzeile
2. Erstellen von leeren Partikel-Sets
3. for-loop
4. Hypothetischen Zustand $x_t^{[m]}$ des Partikels aus letztem Zustand $x_{t-1}^{[m]}$ und Control-Input u_t bestimmen.
   Dieser Schritt beinhaltet das sampeln aus der Zustands-Übergangs-Verteilung

$p(x_t|u_t, x_{t-1})$. Um diesen Schritt implementieren zu können, muss es möglich sein, von dieser Verteilung zu sampeln.

Die Wahrscheinlichkeit einer Zustandsschätzung x_t, die in ein Partikel-Set eingebunden wird, soll proportional zu der Bayes-Filter Posterior $bel(x_t)$ sein. Dazu wird das Bewegungsmodell in Abhängigkeit von u_t und $x_{t-1}^{[m]}$ benötigt.

5. Vergeben eines Gewichtes (,,Importance Factor''). Das Gewicht wird verwendet, um die Messung z_t in das Partikel-Set mit einfließen zu lassen. Das Gewicht ist die Wahrscheinlichkeit der Messung z_t bei Partikel $x_t^{[m]}$. Das gewichtete Partikel-Set repräsentiert eine Annäherung an den Bayes Filter Posterior $bel(x_t)$. Dazu wird das Perzeptionsmodell in Abhängigkeit von $x_t^{[m]}$ und m benötigt. Als Startwert wird allen Partikeln das Gewicht $\frac{1}{M}$ gegeben.

6. Partikel mit Gewicht wird in das temporäre Partikels-Set eingegeben

7. end: for-loop

8. Hier beginnt der wahre ,,Trick'' des Partikel-Filters: Das Resampling.

9. Es wird ein Partikel (mit Zurücklegen) aus dem temporären Partikel-Set gezogen.

10. Der gezogene Partikel wird in das neue Partikel-Set $\mathcal{X}_t$ gespeichert

11. end: for-loop

12. Rückgabe der neuen Partikelverteilung

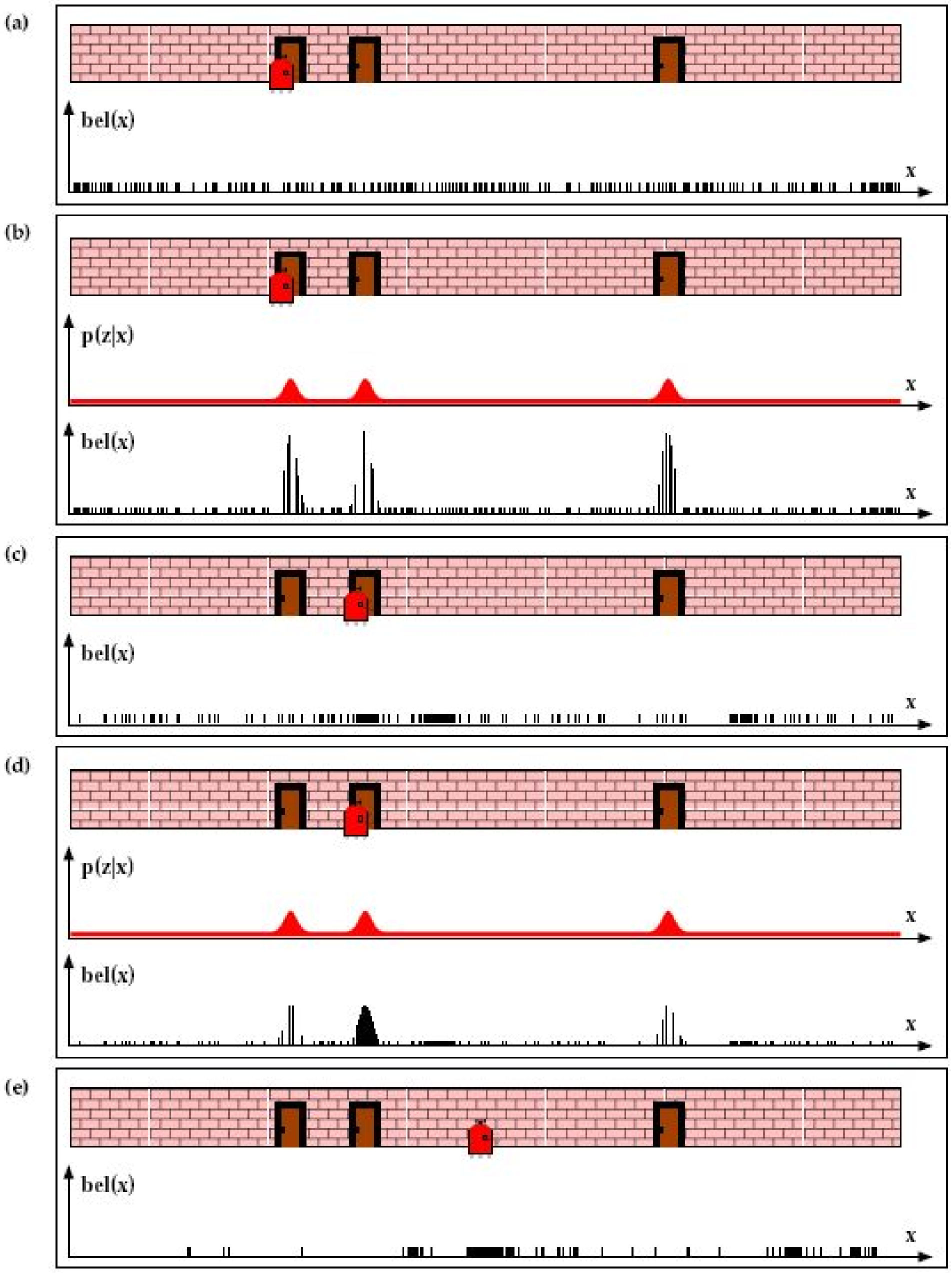

Abbildung 2.1: Partikelfilter für die Lokalisation