

Olasılıksal Robotik

Dr. Öğr. Üyesi Erkan Uslu

11

Eşanlı Konum Belirleme ve Haritalama - SLAM

- Parçacık filtresi SLAM için kullanılabilir mi?
- Mevcut durumda Gaussian Filtreler boyuta göre doğrusal ~ ikinci dereceden arasında ölçeklenebilirken Parçacık Filtreai Üstsel olarak ölçeklenebilir.
- Curse of dimentionality

FastSLAM

- Parçacık filtresi tabanlı bir algoritma
- Parçacıklar baştan itibaren tüm konumları tahmin etmeye çalışır --> Full SLAM

FastSLAM

- Harita özellik noktalarının konumları bağımsız düşük boyutlu EKF'ler ile tahmin edilir
- (EKF SLAM'de tüm harita özellik konumları için tek bir Gaussian ile tahmin edilmeye çalışılıyordu)

FastSLAM

- Her parçacık kendi için özel veri eşleştirme kararına sahiptir --> diğer SLAM'lerden farklı
- Parçacık tabanlı olması ile doğrusal olmayan hareket modelleri rahatlıkla kullanılabilir

FastSLAM - Parçacıklar

	robot path	feature 1	feature 2	...	feature N
Particle $k = 1$	$x_{1:t}^{[1]} = \{(x \ y \ \theta)^T\}_{1:t}^{[1]}$	$\mu_1^{[1]}, \Sigma_1^{[1]}$	$\mu_2^{[1]}, \Sigma_2^{[1]}$...	$\mu_N^{[1]}, \Sigma_N^{[1]}$
Particle $k = 2$	$x_{1:t}^{[2]} = \{(x \ y \ \theta)^T\}_{1:t}^{[2]}$	$\mu_1^{[2]}, \Sigma_1^{[2]}$	$\mu_2^{[2]}, \Sigma_2^{[2]}$...	$\mu_N^{[2]}, \Sigma_N^{[2]}$
		\vdots			
Particle $k = M$	$x_{1:t}^{[M]} = \{(x \ y \ \theta)^T\}_{1:t}^{[M]}$	$\mu_1^{[M]}, \Sigma_1^{[M]}$	$\mu_2^{[M]}, \Sigma_2^{[M]}$...	$\mu_N^{[M]}, \Sigma_N^{[M]}$

FastSLAM – Algoritma Adımları

- Do the following M times:
 - **Retrieval.** Retrieve a pose $x_{t-1}^{[k]}$ from the particle set Y_{t-1} .
 - **Prediction.** Sample a new pose $x_t^{[k]} \sim p(x_t \mid x_{t-1}^{[k]}, u_t)$.
 - **Measurement update.** For each observed feature z_t^i identify the correspondence j for the measurement z_t^i , and incorporate the measurement z_t^i into the corresponding EKF, by updating the mean $\mu_{j,t}^{[k]}$ and covariance $\Sigma_{j,t}^{[k]}$.
 - **Importance weight.** Calculate the importance weight $w^{[k]}$ for the new particle.
- **Resampling.** Sample, with replacement, M particles, where each particle is sampled with a probability proportional to $w^{[k]}$.

SLAM – FastSLAM

- FastSLAM ile Full SLAM problemi $N+1$ posterior çarpımı olarak ifade edilir

$$p(y_{1:t} | z_{1:t}, u_{1:t}, c_{1:t}) = p(x_{1:t} | z_{1:t}, u_{1:t}, c_{1:t}) \prod_{n=1}^N p(m_n | x_{1:t}, u_{1:t}, c_{1:t})$$

- İlk terim parçacık filtresi ile
- İkinci terim grubu EKF ile yapılır, her özellik noktası için ayrı bir EKF var

FastSLAM – Bilinen veri ilişkilendirmesi

- k. parçacık için hareket modeline göre önceki konum ve hareket komutuna göre mevcut konum için örnek üretilir

$$x_t^{[k]} \sim p(x_t | x_{t-1}^{[k]}, u_t)$$

FastSLAM – Bilinen veri ilişkilendirmesi

- t anındaki ölçümde gözlemlenmemiş özellikler için posteriorlar değiştirilmez

$$\left\langle \mu_{n,t}^{[k]}, \Sigma_{n,t}^{[k]} \right\rangle = \left\langle \mu_{n,t-1}^{[k]}, \Sigma_{n,t-1}^{[k]} \right\rangle$$

FastSLAM – Bilinen veri ilişkilendirmesi

- t anında gözlenmiş özellikler için EKF ile güncelleme yapılır

$$K_t^{[k]} = \Sigma_{c_t, t-1}^{[k]} H_t^{[k]T} (H_t^{[k]} \Sigma_{c_t, t-1}^{[k]} H_t^{[k]T} + Q_t)^{-1}$$

$$\mu_{c_t, t}^{[k]} = \mu_{c_t, t-1}^{[k]} + K_t^{[k]} (z_t - \hat{z}_t^{[k]})$$

$$\Sigma_{c_t, t}^{[k]} = (I - K_t^{[k]} H_t^{[k]}) \Sigma_{c_t, t-1}^{[k]}$$

FastSLAM – Bilinen veri ilişkilendirmesi

- Yeniden örnekleme : geçici parçacıklar için ağırlıklandırmaya göre yapılır.

FastSLAM

```
1:  Algorithm FastSLAM 1.0_known_correspondence( $z_t, c_t, u_t, Y_{t-1}$ ):  
2:      for  $k = 1$  to  $M$  do                                // loop over all particles  
3:          retrieve  $\left\langle x_{t-1}^{[k]}, \left\langle \mu_{1,t-1}^{[k]}, \Sigma_{1,t-1}^{[k]} \right\rangle, \dots, \left\langle \mu_{N,t-1}^{[k]}, \Sigma_{N,t-1}^{[k]} \right\rangle \right\rangle$  from  $Y_{t-1}$   
4:           $x_t^{[k]} \sim p(x_t \mid x_{t-1}^{[k]}, u_t)$                 // sample pose  
5:           $j = c_t$                                            // observed feature  
6:          if feature  $j$  never seen before  
7:               $\mu_{j,t}^{[k]} = h^{-1}(z_t, x_t^{[k]})$             // initialize mean  
8:               $H = h'(x_t^{[k]}, \mu_{j,t}^{[k]})$                 // calculate Jacobian  
9:               $\Sigma_{j,t}^{[k]} = H^{-1} Q_t (H^{-1})^T$         // initialize covariance  
10:              $w^{[k]} = p_0$                                 // default importance weight
```

FastSLAM

```
11:     else
12:          $\hat{z} = h(\mu_{j,t-1}^{[k]}, x_t^{[k]})$  // measurement prediction
13:          $H = h'(x_t^{[k]}, \mu_{j,t-1}^{[k]})$  // calculate Jacobian
14:          $Q = H \Sigma_{j,t-1}^{[k]} H^T + Q_t$  // measurement covariance
15:          $K = \Sigma_{j,t-1}^{[k]} H^T Q^{-1}$  // calculate Kalman gain
16:          $\mu_{j,t}^{[k]} = \mu_{j,t-1}^{[k]} + K(z_t - \hat{z})$  // update mean
17:          $\Sigma_{j,t}^{[k]} = (I - K H) \Sigma_{j,t-1}^{[k]}$  // update covariance
18:          $w^{[k]} = |2\pi Q|^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (z_t - \hat{z}_n)^T \right.$   

            $\left. Q^{-1} (z_t - \hat{z}_n) \right\}$  // importance factor
19:     endif
```

FastSLAM

```
20:      for all other features  $j' \neq j$  do           // unobserved features
21:           $\mu_{j',t}^{[k]} = \mu_{j',t-1}^{[k]}$            // leave unchanged
22:           $\Sigma_{j',t}^{[k]} = \Sigma_{j',t-1}^{[k]}$ 
23:      endfor
24:  endfor

25:   $Y_t = \emptyset$                                      // initialize new particle set
26:  do  $M$  times                                         // resample  $M$  particles
27:      draw random  $k$  with probability  $\propto w^{[k]}$  // resample
28:      add  $\left\langle x_t^{[k]}, \left\langle \mu_{1,t}^{[k]}, \Sigma_{1,t}^{[k]} \right\rangle, \dots, \left\langle \mu_N^{[k]}, \Sigma_N^{[k]} \right\rangle \right\rangle$  to  $Y_t$ 
29:  endfor

30:  return  $Y_t$ 
```