

SMC-menetelmät

Laskennallinen tilastotiede - Harjoitustyö

Lasse Rintakumpu

27 April, 2021

Sisällys

1	Johdanto	1
1.1	Suodinongelma	1
1.2	Historiaa	2
2	Bayesilainen suodin	2
3	Hiukkassuodin-algoritmi	3
3.1	Ehdotusjakauman valinta	3
3.2	Algoritmi	3
3.3	Marginaalijakauma	4
3.4	Aikakompleksisuus	4
4	Paikannusesimerkki	4
5	Lopuksi	4
6	Lähteet	4

1 Johdanto

SMC-menetelmät (Sequential Monte Carlo -menetelmät, tunnetaan myös nimellä hiukkassuotimet) ovat joukko 90-luvulta eteenpäin kehitettyjä algoritmeja ja tilastollisia menetelmiä, joiden avulla voidaan approksimoida ns. suodinongelmaa. Tämän tutkielman tavoitteena on esittää pääpiirteittään SMC-menetelmien teoria sekä joitakin menetelmäperheeseen kuuluvia algoritmeja. Tutkielman esitykset seuraavat erityisesti LÄHTEET. SMC-menetelmille on lukuisia sovellutuksia fysiikasta robotiikkaan. Tämän tutkielman lopussa tarkastellaan hiukkassuotimen käyttöä paikannussovelluksessa.

1.1 Suodinongelma

Signaalinkäsittelyssä ja tilastotieteessä suodinongelmaksi kutsutaan XXX Hiukkassuotimen tavoitteena on estimoida “piilossa olevien” tilojen posteriorijakauma havaintojen perusteella. Lisäksi tiedetään miten havaitut muuttujat kytkeytyvät “piilossa oleviin” muuttujiin sekä osaamme sanoa jotain tilamuuttujien todennäköisyyksistä. Signaalinkäsittelyssä tätä kutsutaan suodinongelmaksi. Ongelmaa havainnollistaa kaavio (1).

In the theory of stochastic processes, the filtering problem is a mathematical model for a number of state estimation problems in signal processing and related fields. The general idea is to establish a “best estimate” for the true value of some system from an incomplete, potentially noisy set of observations on that system. Hiukkassuotimet (*particle filters*, tunnetaan myös nimellä sekventiaaliset Monte Carlo -menetelmät,

sequential Monte Carlo, SMC) ovat joukko 90-luvulta eteenpäin kehitettyjä menetelmiä, joiden avulla voidaan approksimoida epälineaarista Bayes-rekursiota / suodinongelmaa.

Menetelmissä käytetään otoksia (kutsutaan myös partikkeleiksi) esittämään jonkin stokastisen prosessin posteriorijakaumaa, kun vain osa havainnoista tunnetaan ja/tai havaintoihin liittyy kohinaa.

$$X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow \dots \text{ piilossa olevat tilat} \quad (1)$$

$$\downarrow \quad \downarrow \quad \downarrow \quad (2)$$

$$Y_1 \rightarrow Y_2 \rightarrow Y_3 \rightarrow \dots \text{ havainnot.} \quad (3)$$

Hiukkassuodin estimoi *sekventiaalisesti* tilojen X_k arvot minä hyvänsä ajan hetkellä k , kun ainoastaan prosessi Y_1, \dots, Y_k tunnetaan. Estimaatit saadaan posteriorijakaumasta $p(x_k|y_1, y_2, \dots, y_k)$, jolle siis muodostetaan Bayesilaisittain approksimaatio havaintojen pohjalta.

Merkitään piilossa olevan prosessin tilaa ajanhetkellä k x_k ja havaittua prosessia y_k . Aika-avaruus on diskreetti. Hiukkassuodinta varten määritellään prosesseille mallit

$$x_{k+1} \sim p(x_{k+1}|x_k)$$

$$y_k \sim p(y_k|x_k),$$

joiden pohjalta halutaan laskea posteriorijakauma $p(x_k|y_{1:k})$. Nämä vastaavat aiemmin määriteltyä ehtoa "tiedetään miten havaitut muuttujat kytkeytyvät"piilossa oleviin" muuttujiin (y_k) sekä osaamme sanoa jotain tilamuuttujien todennäköisyyksistä (x_{k+1})".

1.2 Historiaa

Suodinongelma on ... ratkaisuja linearisoinnista. Ensimmäisen Bayesilaisen/MC-ratkaisun esittivät Ero tähän esitykseen. Alla käytetään modernia esitystä, jonka suurin ero aiempiin on

2 Bayesilainen suodin

Bayesilainen suodin on XXX. Tässä esitety versio on XXX. Bayesilainen ratkaisu posteriorijakaumalle saadaan seuraavalla rekursiolla (käydään läpi jokaiselle ajanhetkelle $k = 1, \dots, t$). Lasketaan ensin

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})}, \quad (4)$$

joka saadaan suoraan Bayesin kaavasta $P(A|B) = P(B|A)P(A)/P(B)$. Normalisointivakio lasketaan integraalina

$$p(y_k|y_{1:k-1}) = \int_{\mathbb{R}^{n_x}} p(y_k|x_k)p(x_k|y_{1:k-1}) dx_k, \quad (5)$$

joka saadaan kokonaistodennäköisyyskaavasta $P(A) = \mathbb{E}[P(A|X)] = \int_{-\infty}^{\infty} P(A|X=x)f_X(x) dx$. Merkintä \mathbb{R}^{n_x} vastaa tässä piilossa olevien muuttujien dimensiota n .

Lopuksi lasketaan päivitysaskel ajalle, joka saadaan edelleen kokonaistodennäköisyydellä

$$p(x_{k+1}|y_{1:k}) = \int_{\mathbb{R}^{n_x}} p(x_{k+1}|x_k)p(x_k|y_{1:k}) dx_k. \quad (6)$$

Rekursiion avulla voidaan siis laskea $p(x_k|y_{1:k})$ käymällä rekursio läpi k kertaa.

3 Hiukkassuodin-algoritmi

Tässä luvussa esitetään algoritmi, joka toteuttaa yllä esitetyn Bayesilaisen suotimen. Luku koostuu XXX algoritmi XX ennen ehdotusjakauman valinta (optimaalinen) lopuksi käsitellään algoritmin marginaalijakaumaa ja aikakompleksisuutta. Esitetään myös ONLOGN-versio algoritmista.

3.1 Ehdotusjakauman valinta

Ennen algoritmin suorittamista valitaan ehdotusjakauma $q(x_{k+1}|x_{1:k}, y_{k+1})$, uudelleenotantamenetelmä sekä partikkelien määrä N . Ehdotusjakauman valintaa ei käsitellä tässä, mutta valinta voidaan tehdä optimaalisesti. Yksinkertaisin valinta on $q(x_{k+1}|x_{1:k}, y_{k+1}) = p(x_{k+1}|x_{1:k})$.

3.2 Algoritmi

Alla käytetään notaatiota x_k^i , joka tarkoittaa, että tila x_k käy ajanhetkellä k gridin pisteessä x^i . Notaatiota tarvitaan, koska hiukkassuotimessa läpikäytävä gridi muuttuu ajan funktiona.

Alustus: Generoidaan $x_1^i \sim p_{x_0}$ missä $i = 1, \dots, N$ ja jokainen partikkeli saa saman painon $w_{1|0}^i = 1/N$. Jokaiselle ajanhetkelle $k = 1, 2, \dots, t$ toistetaan seuraava laskenta.

Vaihe 1: Päivitetään havainnot. Jokaiselle partikkelille $i = 1, 2, \dots, N$ lasketaan painot

$$w_{k|k}^i = \frac{1}{c_k} w_{k|k-1}^i p(y_k | x_k^i),$$

missä normalisointipaino on

$$c_k = \sum_{i=1}^N w_{k|k-1}^i p(y_k | x_k^i).$$

Vaihe 2: Estimoidaan p . Lasketaan tiheydelle approksimaatio

$$\hat{p}(x_{1:k} | y_{1:k}) = \sum_{i=1}^N w_{k|k}^i \delta(x_{1:k} - x_{1:k}^i),$$

missä $\delta(x)$ on Diracin deltafunktio.

Vaihe 3: Valinnainen uudelleenotanta. Otetaan uudet N otosta palauttaen joukosta $\{x_{1:k}^i\}_{i=1}^N$, missä otoksen i todennäköisyys on $w_{k|k}^i$.

Uudelleenotantaa tarvitaan, koska ilman sitä painot alkavat keskittyä muutaman ajanhetken jälkeen tietyille partikkeleille. Uudelleenotantaa ei kuitenkaan kannata aloittaa liian aikaisin, koska se lisää satunnaisotannon epävarmuutta.

Vaihe 4: Aikapäivitys, jos $k < t$. Luodaan ennusteet partikkeleille ehdotusjakaumasta

$$x_{k+1}^i \sim q(x_{k+1} | x_k^i, y_{k+1})$$

ja päivitetään myös partikkelien painot tärkeytysotannalla sen mukaan kuinka todennäköisiä partikkelien ennusteet ovat

$$w_{k+1|k}^i = w_{k|k}^i \frac{p(x_{k+1}^i | x_k^i)}{q(x_{k+1}^i | x_k^i, y_{k+1})}.$$

Palataan takaisin **vaiheeseen 1**. Jatketaan kunnes $k = t$. Kun $N \rightarrow \infty$ algoritmile pätee $\hat{p}(x_{1:k} | y_{1:k}) \xrightarrow{a.s.} p(x_{1:k} | y_{1:k})$.

3.3 Marginaalijakauma

Edellä kuvattu algoritmi tuottaa approksimaation koko prosessin posteriorijakaumalle $p(x_{1:k}|y_{1:k})$. Jos halutaan tietää ainoastaan posteriorijakauman $p(x_k|y_{1:k})$ estimaatti, voidaan käyttää ainoastaan viimeisestä tilasta x_k^i laskettua estimaattia

$$\hat{p}(x_k|y_{1:k}) = \sum_{i=1}^N w_{k|k}^i \delta(x_k - x_k^i).$$

Toinen vaihtoehto on käyttää laskennassa tärkeytyspainoa

$$w_{k+1|k}^i = \frac{\sum_{j=1}^N w_{k|k}^j p(x_{k+1}^i | x_k^j)}{q(x_{k+1}^i | x_k^i, y_{k+1})}$$

yllä esitetyn sijaan. Tällöin jokaisessa aikapäivitysaskeleessa lasketaan painot kaikkien mahdollisten tila-aika-avaruuspolkujen yli. Samoin kuin uudelleenotanta tämä pienentää painojen varianssia.

3.4 Aikakompleksisuus

Menetelmä kuitenkin lisää algoritmin aikakompleksisuutta $\mathcal{O}(N) \rightarrow \mathcal{O}(N^2)$, joten isoilla arvoilla N pitää käyttää jotakin toista versiota algoritmista. Esimerkiksi $\mathcal{O}(N \log(N))$ versio tästä marginaalihiukkassuotimesta on olemassa.

4 Paikannusesimerkki

Esimerkissä analysoidaan ja vertaillaan myös eri versioiden suorituskykyä.

5 Lopuksi

6 Lähteet

- Dahlin & Schön (2019): **Getting Started with Particle Metropolis-Hastings for Inference in Nonlinear Dynamical Models**. <https://arxiv.org/pdf/1511.01707.pdf>.
- Gordon, Salmond & Smith (1993): **Novel approach to nonlinear/non-Gaussian Bayesian state estimation**. <http://www.irisa.fr/aspi/legland/ref/gordon93a.pdf>.
- Gustafsson (2010): **Particle Filter Theory and Practice with Positioning Applications**. <https://ieeexplore.ieee.org/document/5546308>.
- Särkkä (2013): **Bayesian Filtering and Smoothing**. Cambridge University Press. https://users.aalto.fi/~ssarkka/pub/cup_book_online_20131111.pdf.