

Particle Filters en Niet-Lineair Schatten

Yvo Boers

Thales Nederland B.V.

Abstract

In dit artikel wordt een nieuwe revolutionaire methode voor het schatten van grootheden in niet lineaire dynamische systemen behandeld. Deze methode, '*particle filtering*' genaamd, maakt het mogelijk om een heel scala aan niet lineaire schattingsproblemen nagenoeg exact op te lossen. De kern van de oplossing is een effectieve op Monte Carlo simulatie gebaseerde implementatie van de regel van Bayes voor dynamische systemen. Toepassingen van particle filters zijn onder meer: het volgen en schatten van de positie van objecten (bijvoorbeeld vliegtuigen ten behoeve van luchtverkeersleiding) op grond van radar metingen, het schatten van de positie van een auto op grond van GPS metingen (denk aan een navigatie systeem), navigatie voor vliegtuigen op grond van hoogte metingen, ook zijn er toepassingen in de financiële wiskunde en in de medische signaalverwerking. Verder zal worden aangetoond dat met deze nieuwe methode zaken als het meenemen van randvoorwaarden op een hele natuurlijke manier exact kunnen worden ingepast, iets wat met klassieke methoden niet of slechts bij benadering mogelijk is.

1 Introductie

In dit artikel zullen we een relatief nieuwe methode of techniek, genaamd particle filtering, behandelen. Een filter probleem kan (wiskundig gezien) vrij eenvoudig geformuleerd worden. Echter voordat we dit doen zullen we eerst enige achtergrond geven alsmede enkele toepassingen en een kort historisch overzicht.

Een *filtering probleem* is een probleem waarbij interne variabelen, toestanden genaamd, van een dynamisch systeem, geschat worden op grond van metingen aan het systeem. Het dynamische systeem wordt doorgaans beschreven door middel van een stelsel van differentiaal of differentie vergelijkingen. Als voorbeeld van zo'n dynamisch systeem kunt u denken bewegingsvergelijkingen voor een object. De meetvergelijking is een statische vergelijking die een relatie aangeeft tussen de toestand van het systeem en de gemeten grootheden. De relatie is niet exact, in de zin dat een meting nooit perfect is en er meetruis aanwezig is. Weer denkend aan het bewegende object, zou u bij een meetvergelijking kunnen denken aan een radar als sensor. Deze radar levert dan metingen als afstand en bakshoek en eventueel ook nog elevatiehoek en dopplersnelheid. Een en ander is grafisch weergegeven in figuur 1.

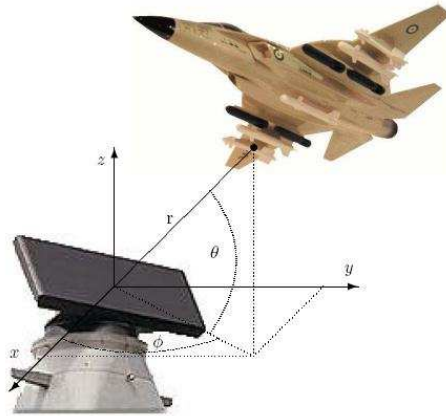


Figure 1: Radar meting aan een bewegend object

Zoals eerder genoemd in de samenvatting zijn er vele toepassingen van het schatten van de toestand in een dynamisch systeem. Het hierboven genoemde voorbeeld is er slechts een van velen.

Het doel van filteren in de context van dit artikel is het extraheren van informatie over de huidige toestand van het systeem gegeven de meet historie. Omdat we te maken hebben met stochastische systemen, namelijk differentiaal of differentie vergelijkingen (deels) onder invloed van ruis is de toestand van zo'n systeem een stochastische variabele. De *filtering dichtheid* representeert de informatie met betrekking tot deze stochastische variabele geconditioneerd op de meet historie. Deze dichtheid wordt ook wel de a posteriori dichtheid genoemd. Het filtering probleem kan nu worden gezien als het probleem deze dichtheid te reconstrueren op basis van gemeten data. Als deze dichtheid eenmaal geconstrueerd is, op wat voor wijze dan ook, is het vrij eenvoudig om (punt) schatters van de toestandvariabelen te verkrijgen. De meest voor de hand liggende en gebruikte schatter is het gemiddelde, maar ook andere puntschatters kunnen worden berekend.

In het algemeen bestaat voor het zojuist beschreven filter probleem geen gesloten of analytische oplossing. Voor een heel beperkte klasse van systemen bestaat deze wel. Voor lineaire systemen onder invloed van Gaussisch verdeelde ruis bestaat er een gesloten analytische recursief uit te rekenen oplossing voor het filter probleem. Voor deze klasse van systemen wordt de oplossing van het filtering probleem beschreven door een Gaussische verdeling. Het *Kalman filter* beschrijft recursieve vergelijkingen voor het gemiddelde en de covariantie als functie van de tijd. We merken hier op dat dit gemiddelde en deze covariantie de Gaussische a posteriori verdeling volledig karakteriseert. Als gezegd bestaat er voor het algemene probleem geen gesloten oplossing. Een 'oplossingsroute' die vaak bewandeld wordt is om het niet lineaire systeem te lineariseren en vervolgens voor het gelineariseerde systeem het Kalman filter toe te passen. Dit is echter een sub-optimale oplossing, die soms redelijk tot goede resultaten oplevert, maar die soms ook tot dramatisch slechte oplossingen kan leiden.

2 Systeem en probleem beschrijvingen

We zullen nu een wat meer formele beschrijving geven van de klasse van systemen, die we zullen beschouwen als ook van het filtering probleem.

Als eerder gezegd concentreren we ons op dynamische systemen met waaraan gemeten wordt. Formeel kunnen we zo'n systeem als volgt weergeven:

$$x_{t+1} = f(x_t, w_t), \quad t \in \mathbb{N} \quad (1)$$

$$z_t = h(x_t, v_t), \quad t \in \mathbb{N} \quad (2)$$

Dit is een beschrijving van een dynamisch systeem in discrete tijd. Vergelijking (1) is de dynamische vergelijking, in dit geval een differentie vergelijking en vergelijking (2) is de meetvergelijking, een algebraïsche vergelijking, die een relatie legt tussen de toestandsgrootheid $x_t \in \mathbb{R}^n$ en de gemeten grootheid $z_t \in \mathbb{R}^n$. In het hiervoor beschreven radar voorbeeld, zouden bijvoorbeeld de 3D positie en snelheid de toestand van het systeem kunnen zijn. De gemeten grootheden zouden bijvoorbeeld afstand en hoek kunnen zijn.

De grootheden w_t en v_t en zijn respectievelijk process- en meetruis. Meetruis is aanwezig in bovenstaande beschrijving omdat in het algemeen sensoren of meetsystemen niet perfect zijn en er een meetfout aanwezig zal zijn. De process ruis geeft aan dat de dynamische vergelijking niet deterministisch is, maar verstoord wordt door een random process. In de modellering en beschrijving van systemen voor bewegingsvergelijkingen van objecten voor radar systemen wordt vaak uitgegaan van een eenparige rechtlijnige beweging, met een random versnelling.

Voor systemen zoals hier beschreven meten we dus bepaalde grootheden, terwijl we eigenlijk geïnteresseerd zijn in andere grootheden (de toestanden). Omdat het beschreven systeem stochastisch van aard is kunnen we slechts verwachten om een beschrijving van de toestanden te verkrijgen in stochastische zin.

Hier komt het zogenaamde filter probleem kijken. Noemen we de meet historie Z_t , waarbij $Z_t = \{z_0, z_1, \dots, z_t\}$ dan bestaat het **filter probleem** uit het verkrijgen van de kansdichtheid:

$$p(x_t | Z_t) \quad (3)$$

Deze dichtheid wordt wel de filter dichtheid of a posteriori dichtheid genoemd. Als we deze kansdichtheid zouden hebben, zouden we zogenaamde puntschatters voor de huidige toestand kunnen construeren. De meest bekende puntschatter is het gemiddelde, welke gelijk is aan de minimum variantie schatter. Deze puntschatter is gelijk aan:

$$x_t^{GEM} = \int x_t p(x_t | Z_t) dx_t \quad (4)$$

Een andere veel gebruikte puntschatter is de zogenaamde Maximum A Posteriori (MAP)schatter. Deze wordt gegeven door:

$$x_t^{MAP} = \arg \max_{x_t} p(x_t | Z_t) \quad (5)$$

Er is een klassieke methode die bijvoorbeeld voor een lineair systeem onder invloed van Gaussische proces- en meetruis de a posteriori dichtheid (recursief) uitrekent, dit filter is het Kalman filter, genoemd naar zijn uitvinder, zie [1]. Dit filter beschrijft de a posteriori verdeling in termen van een gemiddelde en covariantie, wat voldoende is in het geval van een lineair Gaussisch systeem. Het lineaire Gaussische geval is ook eigenlijk de enige systeem beschrijving waarvoor zo'n mooie gesloten uitdrukking bestaat voor de a posteriori kansdichtheid en dus voor de oplossing van het filter probleem.

Een veel gevolgde methode of aanpak is dan ook dat systemen, die niet aan deze beschrijving voldoen eerst benaderd worden door een systeem dat hier wel aan voldoet. Voor niet lineaire systemen wordt dit bijvoorbeeld gedaan door linearisatie. Op deze manier wordt het originele filtering probleem, dat niet analytisch oplosbaar is dus eerst omgevormd tot een simpeler probleem, waarvoor zo'n oplossing wel bestaat. Dit is een praktische methode die echter vele vragen oproept en zeker niet in alle gevallen tot (gewenste) oplossingen leidt. In bepaalde gevallen is de geschetste methodiek ook helemaal niet toepasbaar. Men kan hierbij bijvoorbeeld denken aan systemen, die niet gelineariseerd kunnen worden omdat bijvoorbeeld de meetvergelijking niet differentieerbaar is.

We zullen nu een alternatieve methode beschrijven, het particle filter, die *wel* het filtering probleem voor algemene systemen op kan lossen en waarbij de enige benadering die wordt gedaan in het aantal 'particles' en waarvan is bewezen dat de op deze manier verkregen a posteriori verdeling in een bepaalde zin naar de werkelijke a posteriori verdeling convergeert.

3 Het particle filter

We zullen hier kort een beschrijving van het particle filter geven, alsmede enige uitleg.

Laten we eens aannemen dat het fenomeen of het systeem dat we willen bestuderen leidt tot een a posteriori kansdichtheid zoals weergegeven in figuur 2.

Denk bijvoorbeeld maar aan een applicatie waarin de plaats van object bepaald moet worden en de kansdichtheid in figuur 2, de kansdichtheid voor die plaats, gegeven alle metingen tot op dit moment is. Als we nu een Kalman filter gebruikt zouden hebben (als dit al mogelijk zou zijn!), dan kunnen we op z'n best een beschrijving als in figuur 3 verwachten. Immers het Kalman filter wordt gekarakteriseerd door een gemiddelde en een covariantie. Het moge duidelijk zijn dat deze Kalman beschrijving de werkelijkheid (zoals beschreven in figuur 2 in dit geval niet correct weer kan geven.

Een particle filter benadering van de werkelijke kansdichtheid zou eruit zien als in figuur 4. De werkelijke kansdichtheid wordt benaderd middels een aantal samples of deeltjes (particles), waarbij de locale dichtheid van de particles proportioneel is aan de locale kansdichtheid.

Het particle filter algoritme, dat we zometeen zullen presenteren, propageert de particles in tijd zodanig dat de particles liggen alsof ze getrokken zouden zijn uit de werkelijke, doch onbekende, a posteriori kansdichtheid. Hier

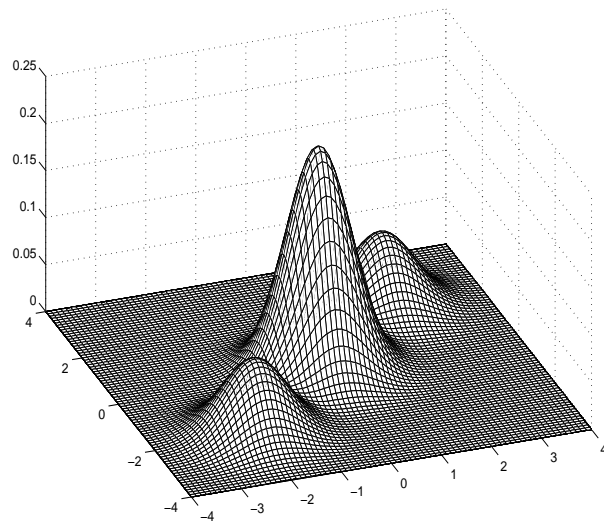


Figure 2: werkelijke a posteriori kansdichtheid

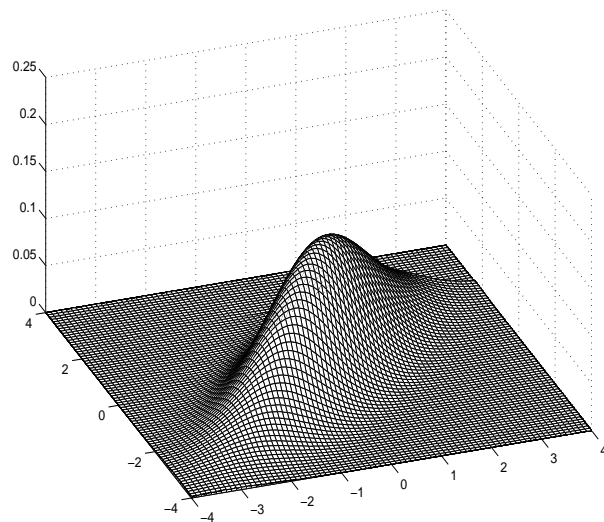


Figure 3: Kalman benadering

moeten we even bij stilstaan, want op zich is dit zeer verwonderlijk en verre van triviaal of voor de hand liggend dat er een algoritme bestaat dat kennelijk in staat is om als het ware trekkingen uit een onbekende kansverdeling te kunnen genereren. Deze trekkingen zijn nu als het ware de oplossing voor ons filter probleem. we kunnen bijvoorbeeld op basis van deze trekkingen een empirisch gemiddelde bepalen.

Het is ook belangrijk dat we ons realiseren dat de *enige* benadering, die we maken, zit in het feit dat het aantal particles eindig is.

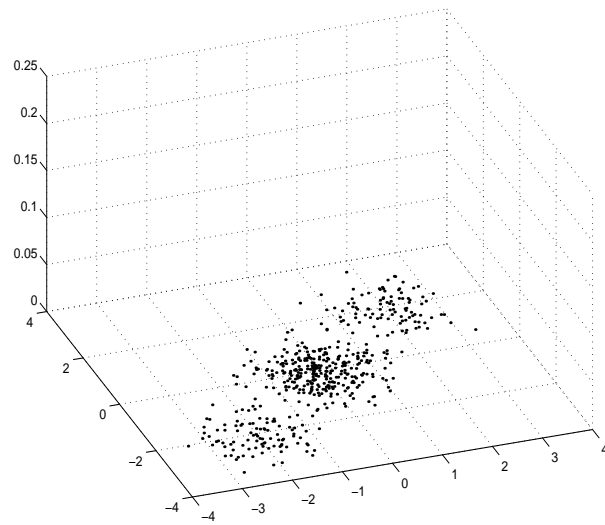


Figure 4: Particle Wolk Approximation

Her particle filter algoritme wordt hier beschreven.

Basis Particle Filter	
I.	Initialisatie: Set $t = 1$ en trek N samples $\{x_0^i\}_{i=1..N}$ volgens de initiële kansdichtheid $p(x_0)$
II.	Predictie: Trek $\{w_{t-1}^i\}_{i=1..N}$ volgens $p_w(w)$ en bereken $\{\tilde{x}_t^i\}_{i=1..N}$ d.m.v. $\tilde{x}_t^i = f(x_{t-1}^i, v_{t-1}^i) \tag{6}$
III.	Update: De meting z_t wordt gebruikt om een likelihood voor elk particle uit te rekenen. $q_t^i = p(z_t \tilde{x}_t^i), \quad i = 1, \dots, N \tag{7}$
IV.	Normalisatie $\tilde{q}_t^i := \frac{q_t^i}{\sum_{i=1}^N w_t^i}, \quad i = 1, \dots, N. \tag{8}$
V.	(Resampling) Trek N keer met teruglegging uit $\{\tilde{x}_t^i, \tilde{q}_t^i\}_{i=1..N} \tag{9}$ and obtain $\{x_t^i\}_{i=1..N}$ to construct $\{x_t^i, N^{-1}\}_{i=1..N} \tag{10}$ $t := t + 1$, goto II.

De notatie $\{x_t^i, q_t^i\}_{i=1..N}$ betekent dat we een verzameling particles x_t^i en gewichten q_t^i hebben.

- In de predictie stap worden alle particles met uniforme gewichten, $\{x_{t-1}^i, N^{-1}\}_{i=1..N}$, gepropageerd volgens de dynamica. Dit resulteert in een nieuwe verzameling particles met uniforme gewichten: $\{\tilde{x}_t^i, N^{-1}\}$.
- Op het moment dat een nieuwe meting beschikbaar komt wordt voor elk particle de likelihood berekend volgens $q_t^i = p(z_t | \tilde{x}_t^i)$, na normalisatie worden de gewichten, \tilde{q}_t^i . Zo krijgen we $\{\tilde{x}_t^i, \tilde{q}_t^i\}_{i=1..N}$. Dit is een verzameling particles met niet-uniforme gewichten.
- De laatste stap is de resampling stap. In deze stap worden particles getrokken volgens de discrete kansdichtheid of wolk $\{\tilde{x}_t^i, \tilde{w}_t^i\}_{i=1..N}$. Dit gebeurt volgens het principe van trekking met teruglegging. Dus sommige particles kunnen meerdere keren worden getrokken. We hebben nu een particle wolk met uniforme gewichten, $\{x_t^i, N^{-1}\}_{i=1..N}$.

Op basis van de particle wolk kunnen nu ook schatters empirisch berekend worden. Het gemiddelde van de particle wolk is een veel gebruikte schatter voor het werkelijke gemiddelde van de verdeling:

$$\hat{x}_t^{GEM} = \frac{1}{N} \sum_{i=1}^N x_t^i$$

Dit particle filter moet worden toegeschreven aan Gordon en co-auteurs, zie [2]. In het verleden, dus pre 1993, zijn dit soort filters al wel voorgesteld, echter de werking bleek niet goed te zijn, in de zin dat na een aantal stappen maar een paar verschillende of zelfs maar een particle overbleef. De sleutel tot een werkend particle filter is de re-sampling stap. Deze bleek cruciaal te zijn.

Het beschreven algoritme is een speciaal geval van een meer algemeen algoritme dat ook wel het bekend staat als het Sampling Importance Resampling (SIR) algoritme, zie [1].

Een van de belangrijkste eigenschappen van het algoritme is dat de particle wolk, zoals aangemaakt door het algoritme, convergeert naar de werkelijke (onbekende!) a posteriori kansdichtheid waarin we zijn geïnteresseerd. Deze convergentie eigenschappen zijn eigenlijk de wiskundige rechtvaardiging voor het gebruik van het particle filter. Meer detail met betrekking tot deze convergentie resultaten zijn o.m. te vinden in [3] en [4].

Het mooie van het hier beschreven algoritme is dat het zeer breed toepasbaar is en dat er eigenlijk maar weinig voorwaarden zijn. De meest belangrijke voorwaarden zijn dat de dynamica bekend is, deze hoeft dus niet lineair te zijn, dat samples getrokken kunnen worden uit volgens de proces ruis, $p_w(w)$, en dat de likelihood geëvalueerd kan worden, wat eigenlijk betekent dat er een meet model moet zijn. Als aan deze (zeer algemene) voorwaarden van kracht zijn is het algoritme toe te passen.

Het is goed dat we ons realiseren dat we een zeer brede klasse van dynamische systemen kunnen behandelen binnen de context van het particle filter. We kunnen zo over weg met niet continue dynamica, en 'wilde' verdelingen voor process- en meetruis. Een zeer belangrijk punt is bijvoorbeeld dat binnen de context van het particle filter zonder meer overweg kunnen randvoorwaarden op de dynamica, denk hierbij bijvoorbeeld aan een auto die zich op de weg en niet naast de weg moet bevinden of een object dat niet sneller kan dan een bepaalde snelheid, of niet harder kan versnellen dan met een bepaalde versnelling, legio andere praktijk voorbeelden zijn denkbaar. Deze informatie is direct en *zonder benadering* mee te nemen.

4 Enkele Voorbeelden

In dit gedeelte zullen we enkele voorbeelden behandelen, die met een particle filter tot mooie oplossingen leiden.

4.1 Sensor netwerk

Het eerste voorbeeld behelst een sensor netwerk. In dit netwerk leveren sensoren binaire informatie over het al dan niet aanwezig zijn van object binnen een bepaalde afstand van deze sensor.

De dynamica en meet vergelijking zijn hier simpel via kansdichtheden te beschrijven. Deze beschrijving is equivalent aan die in sectie 2.

$$x_{t+1} \sim p(x_{t+1}|x_t) \quad (11a)$$

$$z_t \sim p(z_t|x_t), \quad \text{with} \quad t \in \mathbb{N} \quad (11b)$$

x_t is de toestand van het systeem op tijd stip t , z_t is de meting op tijdstip t .

Voor dit systeem bestaat de toestand uit de 2D positie en snelheid.

De kansdichtheid $p(x_{t+1}|t_k)$ wordt volledig bepaald door de dynamica:

$$x_{t+1} = Fx_t + Gw_t \quad (12)$$

met w_t standaard normaal verdeelde ruis en

$$F = \begin{pmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (13)$$

en

$$G = \begin{pmatrix} \frac{1}{2}(\frac{1}{3}a_{max})T^2 & 0 \\ 0 & \frac{1}{2}(\frac{1}{3}a_{max})T^2 \\ \frac{1}{3}a_{max}T & 0 \\ 0 & \frac{1}{3}a_{max}T \end{pmatrix} \quad (14)$$

met a_{max} de maximum versnelling en T de update tijd van het systeem.

De metingen aan het systeem zijn metingen van de individuele sensoren $j \in \{1, \dots, M\}$, met M het aantal sensoren. De meting is binair en is 0 of 1 en de likelihood wordt gegeven door:

$$\begin{aligned} p(z^j|s) = & \mathbf{1}_{R^j}(s)[z^j(p_d^j + (1 - p_d^j)p_{fa}^j) + \\ & + (1 - z^j)(1 - p_d^j)(1 - p_{fa}^j)] + \\ & + (1 - \mathbf{1}_{R^j}(s))[z^j p_{fa}^j + (1 - z^j)(1 - p_{fa}^j)] \end{aligned} \quad (15)$$

met $\mathbf{1}_{R^j}(s)$ de indicator function, die de waarde 1 of 0 aanneemt, en aangeeft of een object binnen de 'sensing range' van de sensor valt of niet. Natuurlijk zijn sensoren niet perfect en dit wordt in deze beschrijving weergegeven door p_d^j and p_{fa}^j , de detectie kans en de vals alarm kans voor sensor j .

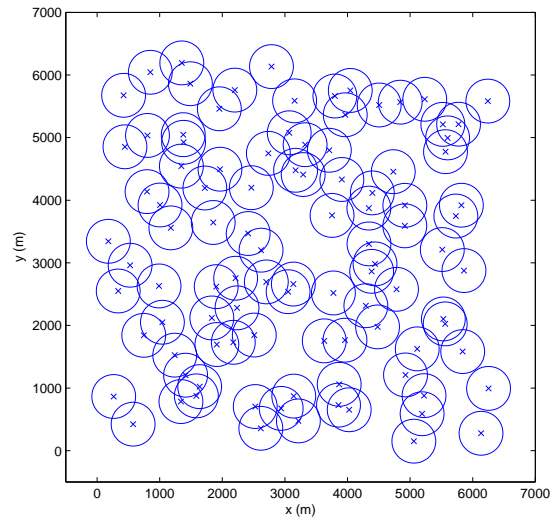


Figure 5: Example of a sensor network layout

De totale likelihood $p(z|s)$ wordt nu verkregen als het product van de individuele likelihoods.

Een voorbeeld van een sensor netwerk layout wordt gegeven in figuur 5. De kruizen geven de sensor locaties aan en de cirkels geven de 'sensing ranges' aan.

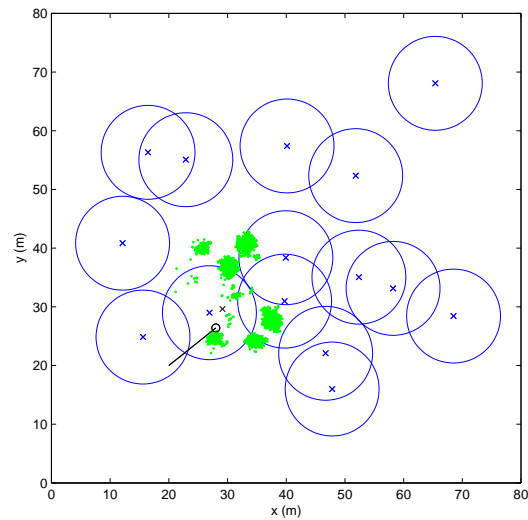


Figure 6: Particle beschrijving van $p(x_k|Z_k)$

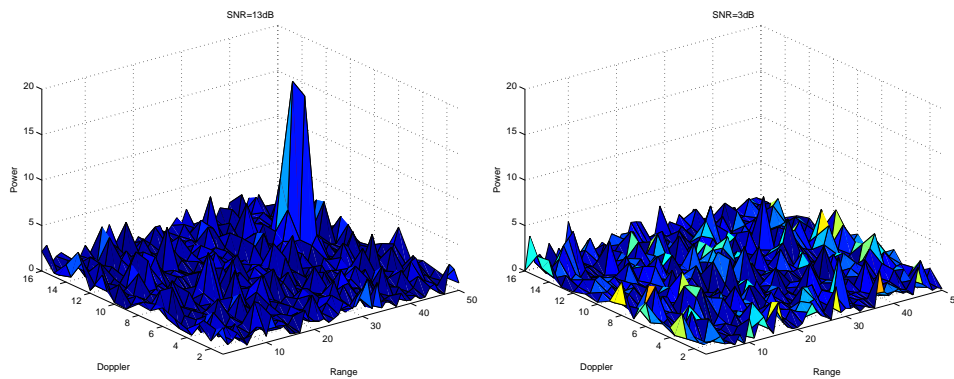
Voor het hier beschreven sensor netwerk probleem is het niet mogelijk om een standaard filter, zoals het Kalman filter, te gebruiken. De reden hiervoor is een niet continue beschrijving van de meetvergelijking. Het particle filter

heeft hier geen probleem mee en de filter dichtheid kan worden gereconstrueerd. Een voorbeeld is gegeven in figuur 6. Wat hier ook goed te zien is, is dat de verdeling in ieder geval niet Gaussisch is en zelfs multimodaal.

Meer informatie over dit specifieke voorbeeld is te vinden in [5].

4.2 Track Before Detect

Dit voorbeeld gaat over het volgen van objecten die zeer zwak zijn. Toepassingen hiervan zijn te vinden in de radar signaal verwerking, maar bijvoorbeeld ook in de biomedische beeldverwerking. In figuur 7 is een voorbeeld geschetst uit de radar signaal verwerking, waar een object in ruis aanwezig is. In het ene geval is het object sterk en is de signaal ruis verhouding (SNR) hoog, in het andere geval hebben we te maken met een zwak object of signaal en is de signaal ruis verhouding laag. In het geval van een sterk object is het object nog wel met het blote oog uit de data te halen, in het geval van het zwakke object is dit niet meer mogelijk. In de radar applicatie zoals hier getoond zijn de cellen zogenaamde range-doppler cellen. Dat wil zeggen dat de cellen gereflecteerde (radar) energie als functie van de afstand (range) en de radiële snelheid (doppler) beschrijven. In een beeldverwerkings applicatie, zouden de cellen pixels met een bepaalde intensiteit kunnen voorstellen.

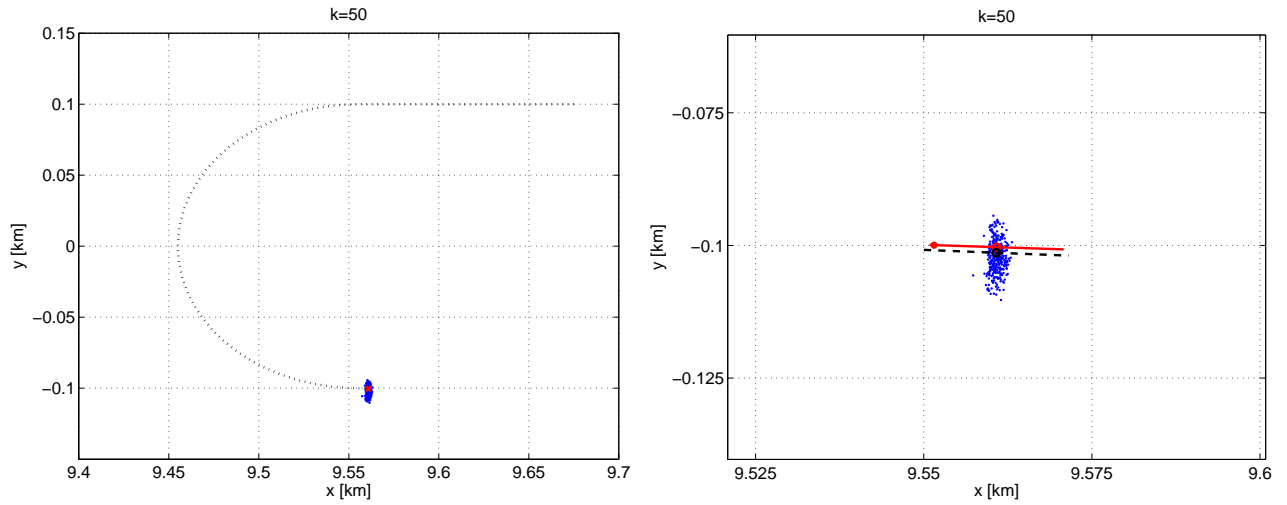


(a) SNR=13 dB. Een hoge SNR maakt dat een object makkelijk te detecteren is.

(b) SNR=3 dB. Een lage SNR maakt dat een object moeilijk te detecteren is.

Figure 7: Voorbeelden van een object (signaal) in ruis bij verschillende SNR.

Voor het volgen van dit soort zwakke objecten is een techniek genaamd 'track-before-detect' bedacht, zie ook [1]. Grofweg kan gezegd worden dat deze techniek de energie, die aanwezig is in het 'plaatje' over meerdere frames integreert en dit zodanig doet dat de juiste cellen (of pixels), in dit geval in het range-doppler domein, met elkaar gecombineerd worden. Als de geïntegreerde energie hoog genoeg is kan eventueel een beslissing genomen worden over de al dan niet aanwezigheid van een object. U kunt zich voorstellen dat het combineren van verschillende cellen over de tijd voor bewegende objecten al heel snel numeriek uit de hand kan lopen. Neem bijvoorbeeld aan dat van tijdstap op tijdstap het object 1 cel op zou kunnen schuiven. Na 10 tijdstappen zouden we dan al 8^{10} (grofweg 1 miljard) mogelijke hypotheses hebben over waar het object zich bevindt en dit ervan uitgaande dat we op het begin tijdstip precies weten in welke cel het object zich bevindt. Het is gebleken dat dit exploreren van de mogelijke



(a) Object trajectorie en de particle wolk op tijdstip $t = 50$.

(b) Detail - particle wolk met de werkelijke (ononderbroken) en geschatte object afmeting (onderbroken).

Figure 8: Scenario voor een 'extended object'

combinaties en het tegelijkertijd integreren van de bijbehorende energie met een particle filter redelijk efficiënt kan worden gedaan. Met redelijk efficiënt doelen we hier op zo'n paar honderd tot paar duizend particles. Een particle stelt in deze applicatie een mogelijke positie van het object voor en het bijbehorende gewicht is een maat voor de waarschijnlijkheid dat het object zich op die plaats bevindt, waarin ook de gereflecteerde energie op die positie is verdisconteerd. De eerste werkende particle gebaseerde algoritmes voor deze applicatie zijn in [6] en [7] beschreven.

Zonder verder in te gaan op modellering details geven we hier enige resultaten van een track-before-detect applicatie voor objecten die niet meer als punt objecten beschouwd kunnen worden, maar die zich over meerdere cellen uitbreiden. In dit geval gaat het om een 1D object, oftewel een object met slechts in een richting een noemenswaardige extensie ten opzichte van de cel grootte. De resultaten hier zijn illustratief en waren mede bedoeld om te laten zien dat het in een particle filter eenvoudig is om de vorm van het object mee te schatten.

Meer detail met betrekking tot dit voorbeeld is te vinden in [8].

5 Conclusies en verder

Dit artikel is bedoeld om u een idee te geven van de werking van een relatief nieuwe methode, die op het gebied van schatten en filteren behoorlijk revolutionair te noemen is. Deze methodiek biedt bepaalde voordelen en heeft ondanks haar korte bestaan al tot enige (werkende) industriële applicaties geleid. Deze techniek en de theory erachter alsmede de applicaties zijn nog volop in ontwikkeling en zullen nog jaren een interessant onderzoeksgebied vormen met een groot aantal open vragen.

Dit artikel is geenzins bedoeld om een volledig overzicht te geven, maar zoals gezegd slechts een idee van de werking en enige mogelijke toepassingen.

De geïnteresseerde lezer vindt in het uitstekende (en overigens tot nu toe ook enige!) tekstboek over dit onderwerp, [1], een veel betere, meer complete en gedetailleerde uiteenzetting over dit onderwerp, met legio andere voorbeelden alsmede ook simpel te implementeren algoritmes in pseudo-code.

Daarnaast geeft de 'Sequential Monte Carlo' website, zie [9], veel informatie en (links naar) artikelen die voor een groot deel vrijelijk kunnen worden gedownload.

6 Dankbetuiging

Dit werk is deels gesponsord door de Nederlandse Organisatie voor Wetenschappelijk Onderzoek (NWO) onder het Casimir programma, contract nummer 018.003.004.

References

- [1] S. Arulampalam, N. Gordon and R. Ristic, "Beyond the Kalman Filter - Particle Filters for Tracking Applications," Boston - London: Artech House 2004.
- [2] N.J. Gordon, D.J. Salmond and A.F.M. Smith. A Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, 140(2), 1993.
- [3] D. Crisan and A. Doucet, A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, 50(3): 736-746, 2002.
- [4] Xiao-Li Hu, T. B. Schumaker and L. Ljung. A Basic Convergence Result for Particle Filtering. *IEEE Transactions on Signal Processing*, 56(4):1337-1348, 2008.
- [5] Y. Boers, J.N. Driessen and L. Schipper. Particle Filter Based Sensor Selection in Binary Sensor Networks. *In proceedings of FUSION 2008*, Cologne, Germany, July 2008.
- [6] D. J. Salmond and H. Birch. A particle filter for track-before-detect. *In proceedings of the American Control Conference*, Arlington, VA, USA, June 2001.
- [7] Y. Boers and J.N. Driessen. Particle Filter based Detection for Tracking. *In proceedings of the American Control Conference*, Arlington, VA, USA, June 2001.
- [8] Y. Boers, J.N. Driessen, J. Torstensson, M. Trieb, R. Karlsson and F. Gustafsson, An Extended Target Particle Filter for a Track Before Detect Application. *IEE Proceedings Radar, Sonar and Navigation*, vol. 153, no. 3, 2006.
- [9] Sequential Monte Carlo Homepage. <http://www.sigproc.eng.cam.ac.uk/smc/>