Dynamical Models for Tracking with the Variable Rate Particle Filter

Pete Bunch and Simon Godsill

Cambridge University Engineering Department
Signal Processing & Communications Lab

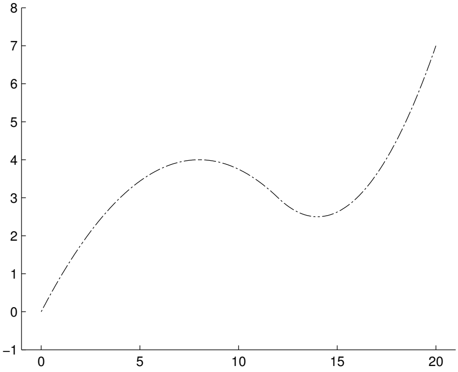
11th July, 2012

What's in store...

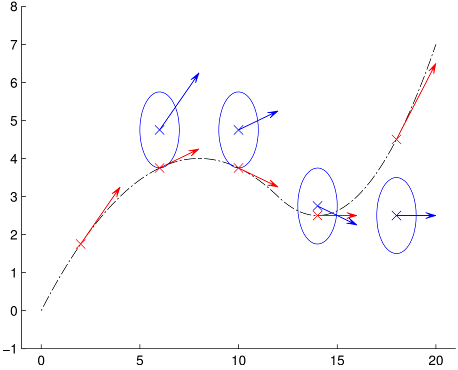
- What is a variable rate model?
- How do we do inference in a variable rate model?
- What dynamical models can we use for tracking with a VRPF?
- What's wrong with them?
- How can make them better?
- Does it work in 3 dimensions?

What is a variable rate model?

Tracking

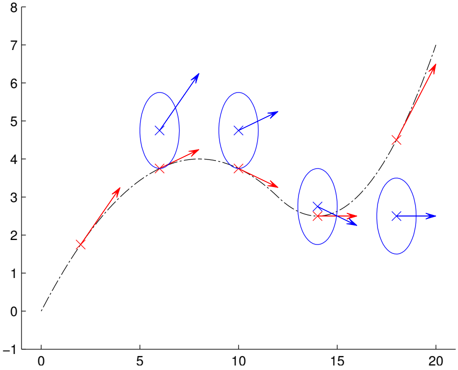


Conventional Hidden Markov Models for Tracking



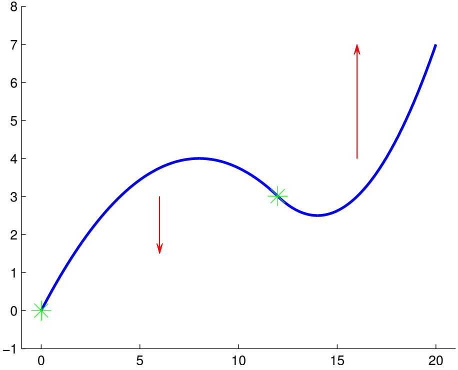
$$x_n = f(x_{n-1}) + w_n$$

Conventional Hidden Markov Models for Tracking



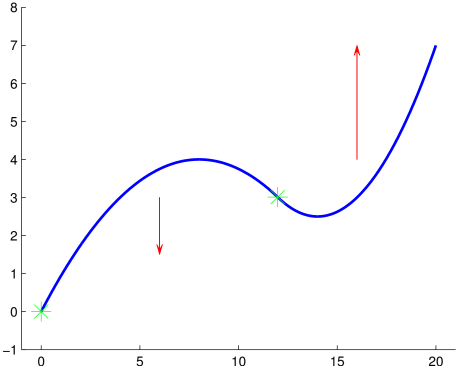
- Discrete-time
- Stochastic Markovian dynamics
- Simple algorithms
- Additional modelling to handle manoeuvres

Variable Rate Models for Tracking



$$x(t) = f(\tau_k, x_k, u_k, t)$$

Variable Rate Models for Tracking



- Continuous-time
- Conditionally-deterministic dynamics
- Handles manoeuvres integrally

How do we do inference in a variable rate model?

The Variable Rate Particle Filter

The Variable Rate Particle Filter

Need to infer:

- Changepoint times, $\{\tau_k\}$
- Motion parameters, $\{u_k\}$

The Variable Rate Particle Filter

Need to infer:

- Changepoint times, $\{\tau_k\}$
- Motion parameters, $\{u_k\}$

Bootstrap particle filter

- Proposal: Add a new changepoint or just extend trajectory
- Weight update: $w_n \propto w_{n-1} p(y_n | x(t_n))$

The Variable Rate Particle Filter

Need to infer:

- Changepoint times, $\{\tau_k\}$
- Motion parameters, $\{u_k\}$

Bootstrap particle filter

- Proposal: Add a new changepoint or just extend trajectory
- Weight update: $w_n \propto w_{n-1} p(y_n | x(t_n))$

Improvements

- Resample-move
- SMC sampler
- RJ-MCMC

What dynamical models can we use for tracking with a VRPF?

VR Tracking Models I: Cartesian Coordinates

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \ddot{x}_1(t) \\ \ddot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a_{1,k} \\ a_{2,k} \end{bmatrix}$$

$$\tau_k \leq t < \tau_{k+1}$$

VR Tracking Models I: Cartesian Coordinates

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \ddot{x}_1(t) \\ \ddot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a_{1,k} \\ a_{2,k} \end{bmatrix}$$

$$\tau_k \leq t < \tau_{k+1}$$

$$\begin{bmatrix} x_1(t) \\ x_2(t) \\ \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(\tau_k) \\ x_2(\tau_k) \\ \dot{x}_1(\tau_k) \\ \dot{x}_2(\tau_k) \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2 & 0 \\ 0 & \frac{1}{2}\Delta t^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} a_{1,k} \\ a_{2,k} \end{bmatrix}$$

$$\Delta t = t - \tau_k$$

VR Tracking Models II: Intrinsic Coordinates

$$
\begin{aligned}
\dot{s}(t) &= a_{T,k} \\
s(t)\dot{\psi}(t) &= a_{N,k} \\
\dot{x}_1(t) &= s(t)\cos(\psi(t)) \\
\dot{x}_2(t) &= s(t)\sin(\psi(t))
\end{aligned}
$$

$$\tau_k \leq t < \tau_{k+1}$$

$$
\begin{aligned}
\dot{s}(t) &= \dot{s}(\tau_k) + a_{T,k}\Delta t \\
\psi(t) &= \psi(\tau_k) + \frac{a_{N,k}}{a_{T,k}} \log\left(\frac{\dot{s}(t)}{\dot{s}(\tau_k)}\right) \\
x(t) &= x(\tau_k) + \frac{\dot{s}(t)^2}{4a_{T,k}^2 + a_{N,k}^2}\left[a_{N,k}\sin(\psi(t)) + 2a_{T,k}\cos(\psi(t))\right] \\
&\quad - \frac{\dot{s}(\tau_k)^2}{4a_{T,k}^2 + a_{N,k}^2}\left[a_{N,k}\sin(\psi(\tau_k)) + 2a_{T,k}\cos(\psi(\tau_k))\right] \\
y(t) &= y(\tau_k) + \frac{\dot{s}(t)^2}{4a_{T,k}^2 + a_{N,k}^2}\left[-a_{N,k}\cos(\psi(t)) + 2a_{T,k}\sin(\psi(t))\right] \\
&\quad - \frac{\dot{s}(\tau_k)^2}{4a_{T,k}^2 + a_{N,k}^2}\left[-a_{N,k}\cos(\psi(\tau_k)) + 2a_{T,k}\sin(\psi(\tau_k))\right]
\end{aligned}
$$

$$\Delta t = t - \tau_k$$

What's wrong with them?

A Problem: Model Deficiency

- 4 state variables governed by 2 motion parameters
- From a given state and time, not all future states are achievable

Problems:

- Poor robustness to model errors
- Forward-backward smoothing does not work

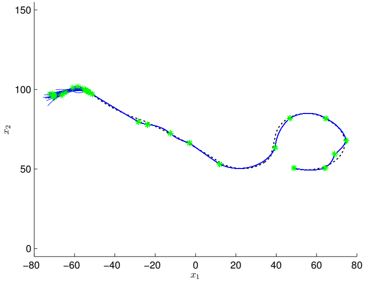
How can make them better?

The Solution: Add More Parameters

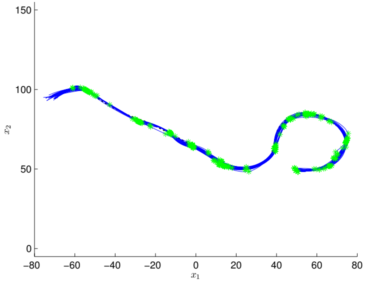
$$\begin{aligned}
\dot{s}(t) &= a_{T,k} \\
s(t)\dot{\psi}(t) &= a_{N,k} \\
\dot{x}_1(t) &= s(t)\cos(\psi(t)) + d_{1,k} \\
\dot{x}_2(t) &= s(t)\sin(\psi(t)) + d_{2,k}
\end{aligned}$$

$$\tau_k \leq t < \tau_{k+1}$$

Variable Rate Smoothing



Filter



Smoother

Does it work in 3 dimensions?

3D Intrinsic Coordinate Tracking

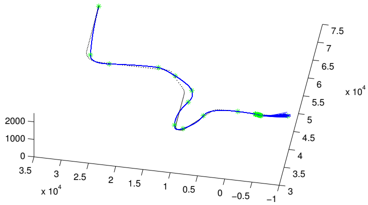
$$
\begin{aligned}
\dot{s}(t) &= a_{T,k} \\
\dot{\mathbf{e}}_T(t) &= \frac{a_{N,k}}{s(t)}\mathbf{e}_N(t) \\
\dot{\mathbf{e}}_N(t) &= -\frac{a_{N,k}}{s(t)}\mathbf{e}_T(t) \\
\dot{\mathbf{e}}_B(t) &= \mathbf{0}.
\end{aligned}
$$

3D Intrinsic Coordinate Tracking

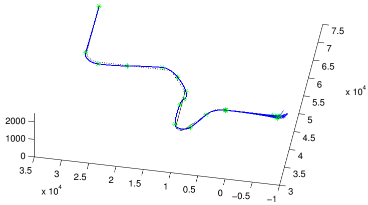
$$
\begin{aligned}
\dot{s}(t) &= \dot{s}(\tau_k) + a_{T,k}\Delta t \\
\begin{bmatrix} \mathbf{e}_T(t)^T \\ \mathbf{e}_N(t)^T \\ \mathbf{e}_B(t)^T \end{bmatrix} &= \begin{bmatrix} \cos(\Delta\psi(t)) & \sin(\Delta\psi(t)) & 0 \\ -\sin(\Delta\psi(t)) & \cos(\Delta\psi(t)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{e}_T(\tau_k)^T \\ \mathbf{e}_N(\tau_k)^T \\ \mathbf{e}_B(\tau_k)^T \end{bmatrix} \\
\Delta\psi(t) &= \frac{a_{N,k}}{a_{T,k}} \log\left(\frac{\dot{s}(t)}{\dot{s}(\tau_k)}\right)
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{v}(t) &= \dot{s}(t)\mathbf{e}_T(t) \\
\mathbf{r}(t) &= \mathbf{r}(\tau_k) + \frac{1}{a_{N,k}^2 + 4a_{T,k}^2} \times \begin{bmatrix} \mathbf{e}_T(\tau_k) & \mathbf{e}_N(\tau_k) \end{bmatrix} \begin{bmatrix} \zeta_1(t) & \zeta_2(t) \\ \zeta_2(t) & -\zeta_1(t) \end{bmatrix} \begin{bmatrix} 2a_{T,k} \\ a_{N,k} \end{bmatrix} \\
\zeta_1(t) &= \cos(\Delta\psi(t))\dot{s}(t)^2 - \dot{s}(\tau_k)^2 \\
\zeta_1(t) &= \sin(\Delta\psi(t))\dot{s}(t)^2
\end{aligned}
$$

3D Intrinsic Coordinate Tracking



Cartesian



Intrinsic

Summary

- Variable rate framework elegantly handles manoeuvres, changepoints and model-switching.
- Variable rate particle filter may be used for inference in this framework.
- Basic VR tracking models are uncontrollable - thwarts smoothing algorithms.
- New augmented model improves estimation and may be used for smoothing.
- Intrinsic coordinate model has been extended to work in 3D - piecewise planar assumption.