Fixed-lag sampling strategies for RBPF SLAM

Kris Beevers

Department of Computer Science
Rensselaer Polytechnic Institute

September 6, 2006

Disclaimer!

I will eventually talk about something new and exciting

- But first there is a lot of background
- And the new stuff is sort of technical
- So I'll only have three slides on the new stuff
- And then I'll just show a bunch of plots
- And you'll just have to sort of trust me that everything is correct
- (Or you can read the paper)
- **But of course**:
  - If you see an equation that is simply too beautiful to let slip by ...
  - Please ask about it!

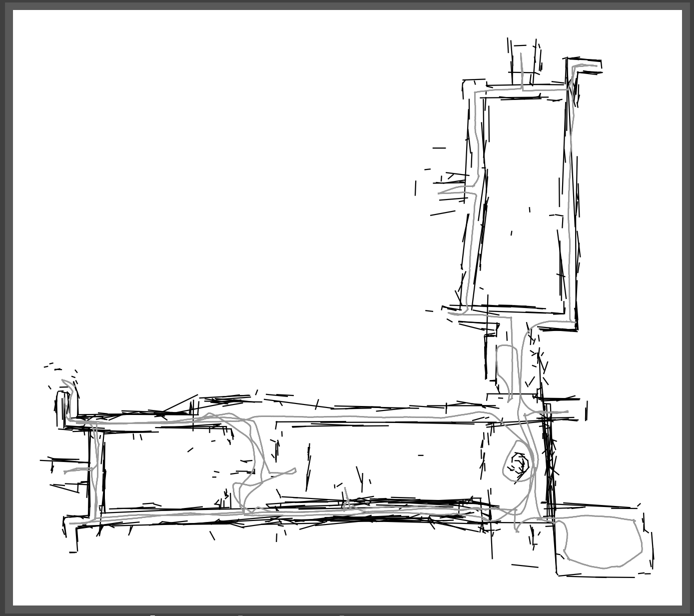
SLAM

SLAM: *simultaneous localization and mapping*

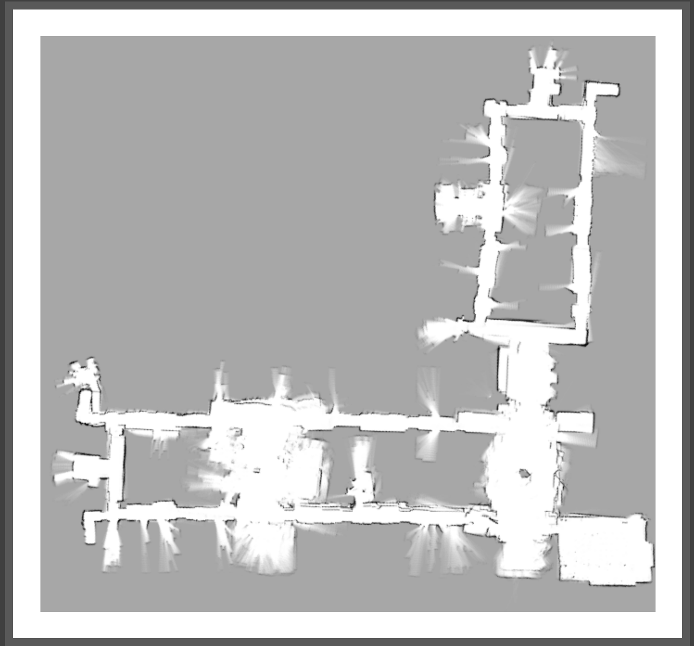
- Concurrently estimate:
  - **map** $\mathbf{x}^m = [\mathbf{x}_1^m \dots \mathbf{x}_n^m]^T$
  - **robot pose** $\mathbf{x}_t^r = [x_t \; y_t \; \theta_t]^T$ (or trajectory $\mathbf{x}_{1:t}^r$)
- Given:
  - **control inputs** $\mathbf{u}_{1:t} = [d_t \; \alpha_t]^T$ (translation, rotation)
  - **observations** $\mathbf{z}_{1:t}$ (e.g., laser, IR, SONAR)
  - **correspondences** $\mathbf{n}_{1:t} : \mathbf{z} \rightarrow \mathbf{x}^m$

$$\boxed{p(\mathbf{x}_{1:t}^r, \mathbf{x}^m | \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t})}$$

raw data (laser + odometry)



landmark map



occupancy map

Basic algorithm (landmark based SLAM)

loop
  Move according to $\mathbf{u}_t$
  Predict new pose $\hat{\mathbf{x}}_t^r$
  Acquire sensor readings and extract features $\mathbf{z}_t$
  Compute correspondences $\mathbf{n}_t$
  Update pose/map estimate $p(\mathbf{x}_t^r, \mathbf{x}^m)$ based on observed landmarks
  Add new landmarks to map
end loop

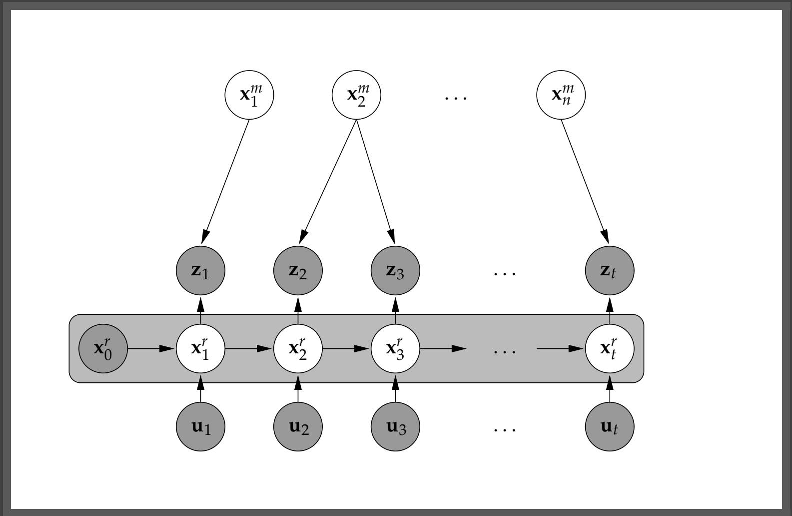
- Typically implemented using Bayesian filtering:

$$\underbrace{p(\mathbf{x}_t | \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t})}_{\text{posterior}} = \eta \underbrace{p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{n}_t)}_{\text{measurement}} \int \underbrace{p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t)}_{\text{motion}} \underbrace{p(\mathbf{x}_{t-1} | \mathbf{u}_{1:t-1}, \mathbf{z}_{1:t-1}, \mathbf{n}_{1:t-1})}_{\text{prior}} \, d\mathbf{x}_{t-1}$$

RBPF SLAM

RBPF: *Rao-Blackwellized particle filtering*

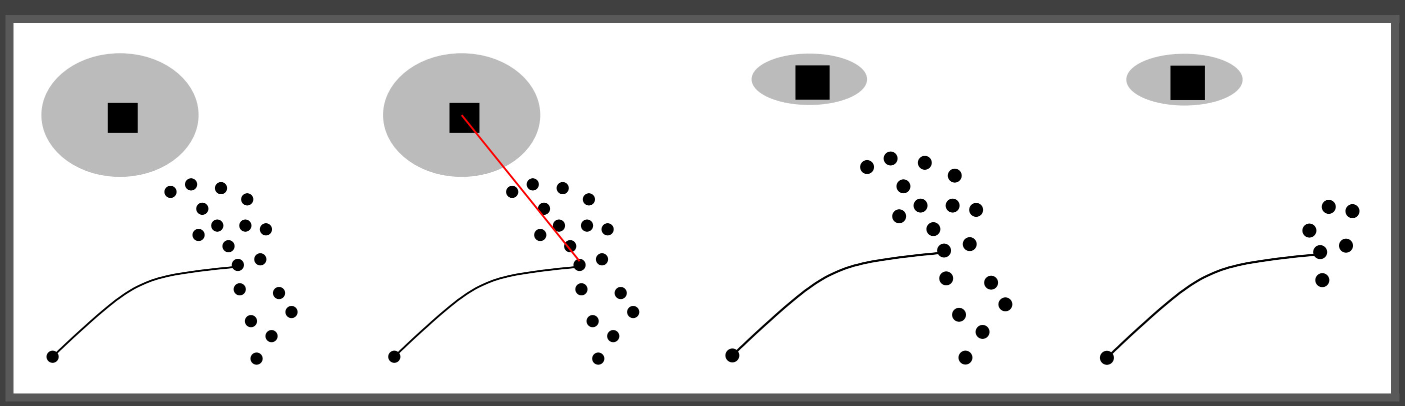
- Compute posterior over **trajectories** and maps
- Markov assumption: landmarks independent conditioned on trajectory



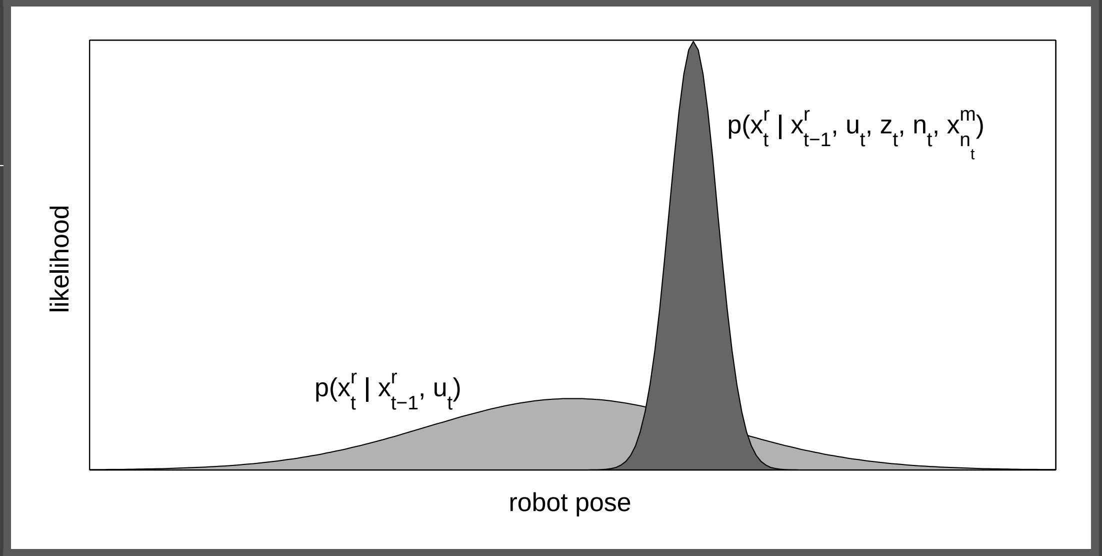
$$p(\mathbf{x}^r_{1:t}, \mathbf{x}^m | \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t}) = \underbrace{p(\mathbf{x}^r_{1:t} | \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t})}_{\text{posterior over trajectories}} \prod_{i=1}^{n} \underbrace{p(\mathbf{x}^m_i | \mathbf{x}^r_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t})}_{\text{posterior over landmark } i}$$

RBPF SLAM algorithm: "FastSLAM 1"

loop
    Move/sense/extract features
    for all particles ϕ^i **do**
        Project forward: $\mathbf{x}_t^{r,i} \sim p(\mathbf{x}_t^r | \mathbf{x}_{t-1}^{r,i}, \mathbf{u}_t)$
        Do data association (compute $\mathbf{n}_t^i$), update map
        Compute weight: $\omega_t^i = \omega_{t-1}^i \times p(\mathbf{z}_t | \mathbf{x}_t^{r,i}, \mathbf{x}^{m,i}, \mathbf{n}_t^i)$
    end for
    Resample (with replacement) according to ω_t^is
end loop



Improved proposal



- Standard RBPF doesn't use $\mathbf{z}_t$ in "proposing" the pose $\mathbf{x}^r_t$
- Typically $p(\mathbf{z}_t|\mathbf{x}^r_t, \mathbf{n}_t, \mathbf{x}^m)$ is much more precise than $p(\mathbf{x}^r_t|\mathbf{x}^r_{t-1}, \mathbf{u}_t)$
- So, only a few samples are highly weighted

This leads to **degeneracies**

- "FastSLAM 2": sample new poses from "improved proposal distribution"

$$\boxed{p(\mathbf{x}^r_t|\mathbf{x}^r_{t-1}, \mathbf{u}_t, \mathbf{z}_t, \mathbf{n}_t, \mathbf{x}^m)}$$

- This is (mostly) where things stand

Rethinking the past

- Key idea: $\mathbf{z}_t$ tells you something about $\mathbf{x}_t^r \ldots$
  - But also about $\mathbf{x}_{t-1}^r, \mathbf{x}_{t-2}^r, \ldots$
- Ideally we should update our belief about every pose in the trajectory
  - In RBPF: draw new samples for the entire trajectory, estimate new maps for the new samples, etc.
  - Big benefit: avoids degeneracies due to resampling
  - Better representation of $p(\mathbf{x}_{1:t}^r | \cdot)$
  - Of course this isn't feasible
- Maybe at least we can do something over a **fixed lag time**
  - Draw new samples for $\mathbf{x}_{t-L+1:t}^r$
  - Update maps from $t - L$ conditioned on new samples

Fixed-lag roughening

- After resampling, apply an MCMC move step to $\{\mathbf{x}^{r,i}_{t-L+1:t}\}$
- Fixed-lag Gibbs sampler for RBPF SLAM:

$$
\begin{aligned}
\mathbf{x}^{r,i}_{t-L+1} &\sim p(\mathbf{x}^{r}_{t-L+1}|\mathbf{x}^{r,i}_{1:t-L,t-L+2:t}, \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t}) \\
&\dots \\
\mathbf{x}^{r,i}_{k} &\sim p(\mathbf{x}^{r}_{k}|\mathbf{x}^{r,i}_{1:k-1,k+1:t}, \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t}) \\
&\dots \\
\mathbf{x}^{r,i}_{t} &\sim p(\mathbf{x}^{r}_{t}|\mathbf{x}^{r,i}_{1:t-1}, \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t})
\end{aligned}
$$

$$
p(\mathbf{x}^{r}_{k}|\mathbf{x}^{r,i}_{1:k-1,k+1:t}, \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t}) =
$$

$$
\eta \int \underbrace{p(\mathbf{z}_k|\mathbf{x}^{r,i}_k, \mathbf{n}_k, \mathbf{x}^m_{\mathbf{n}_k})}_{\text{measurement}} \underbrace{p(\mathbf{x}^m_{\mathbf{n}_k}|\mathbf{x}^{r,i}_{1:k-1,k+1:t}, \mathbf{z}_{1:k-1,k+1:t}, \mathbf{n}_{1:t})}_{\text{landmark}} \, d\mathbf{x}^m_{\mathbf{n}_k}
$$

$$
\underbrace{p(\mathbf{x}^r_k|\mathbf{x}^{r,i}_{k-1}, \mathbf{u}_k)}_{\text{forward}} \underbrace{p(\mathbf{x}^r_k|\mathbf{x}^{r,i}_{k+1}, \mathbf{u}_{k+1})}_{\text{backward}}
$$

Block proposal

- Draw $\{\mathbf{x}^{r,i}_{t-L+1:t}\}$ from joint "L-optimal block proposal" distribution:

$$\boxed{p(\mathbf{x}^r_{t-L+1:t}|\mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t}, \mathbf{x}^{r,i}_{t-L})}$$

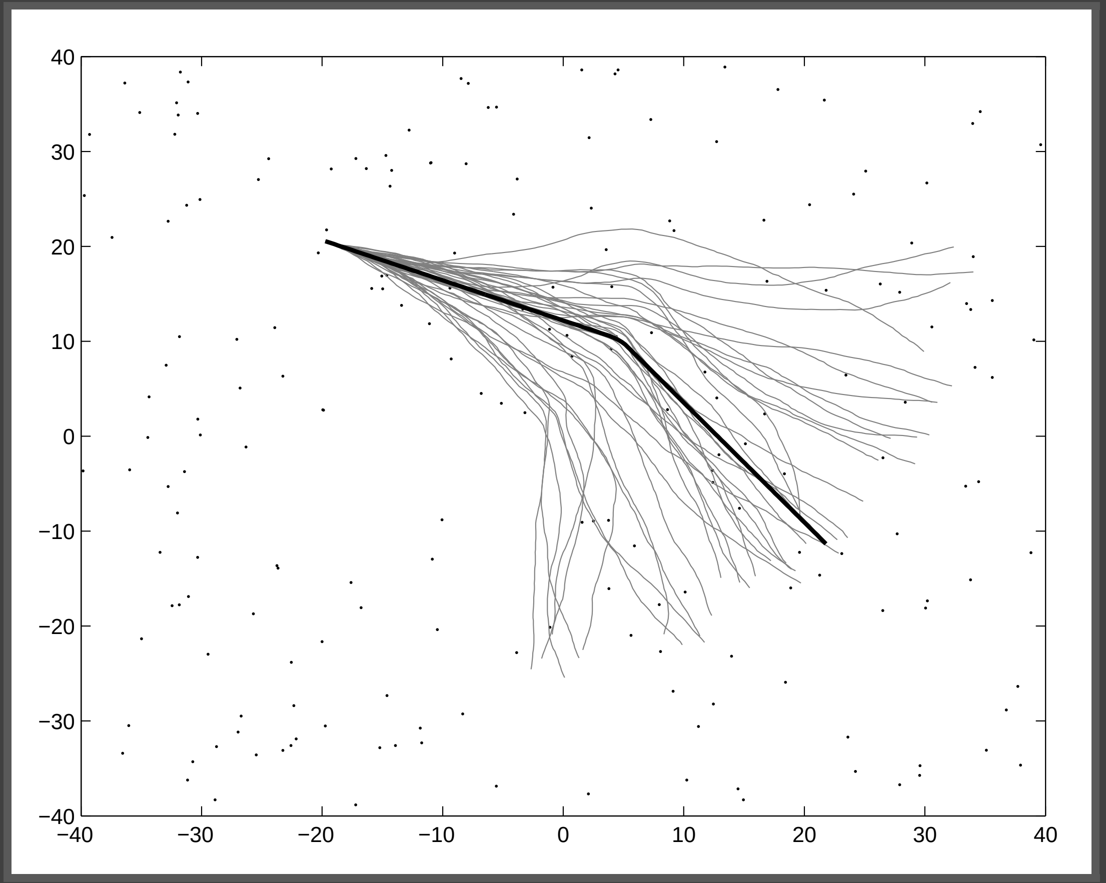
- How to do it: *forward filtering/backward sampling* (Chib, 1996)

$$\underbrace{p(\mathbf{x}^r_k|\mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t}, \mathbf{x}^{r,i}_{t-L}, \mathbf{x}^{r,i}_{k+1:t})}_{\text{sampling distribution}} \propto \underbrace{p(\mathbf{x}^r_k|\mathbf{u}_{1:k}, \mathbf{z}_{1:k}, \mathbf{n}_{1:k}, \mathbf{x}^{r,i}_{t-L})}_{\text{forward filtering}} \underbrace{p(\mathbf{x}^r_{k+1}|\mathbf{x}^{r,i}_k, \mathbf{u}_{k+1})}_{\text{backward model}}$$

  - Filter forward using an EKF
  - Sample $\mathbf{x}^{r,i}_t \sim p(\mathbf{x}^r_t|\mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t}, \mathbf{x}^{r,i}_{t-L})$
  - Compute sampling distribution for $\mathbf{x}^{r,i}_{t-1}$ and sample
  - Continue back to $t - L + 1$

- Need to reweight particles: $\omega^i_t = \omega^i_{t-1}\, p(\mathbf{z}_t|\mathbf{x}^{r,i}_{1:t-L}, \mathbf{u}_{1:t}, \mathbf{z}_{1:t-1}, \mathbf{n}_{1:t})$

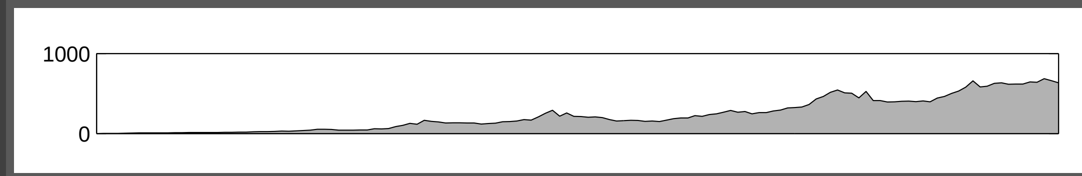
Results: sparse environment



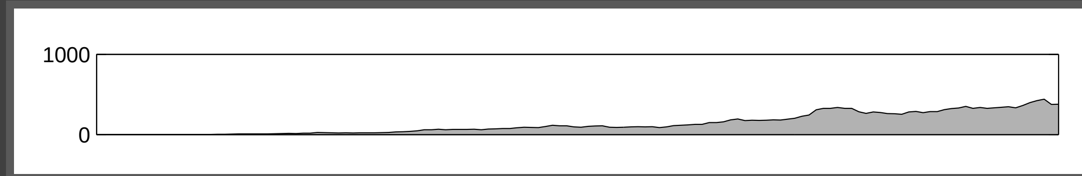
- 27 sec., no loops
- 50 Monte Carlo trials averaged for all results

Norm. est. error sq. (NEES): $(\mathbf{x}_t^r - \hat{\mathbf{x}}_t^r)(\hat{\mathbf{P}}_t^r)^{-1}(\mathbf{x}_t^r - \hat{\mathbf{x}}_t^r)^T$

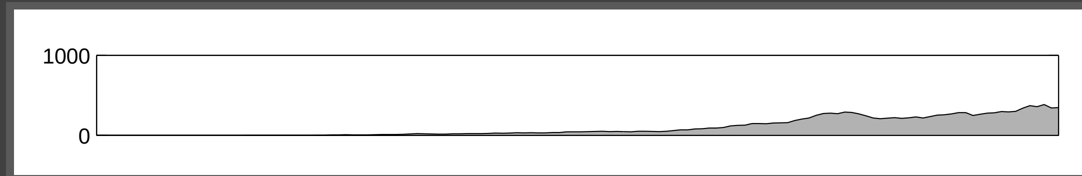
FS2



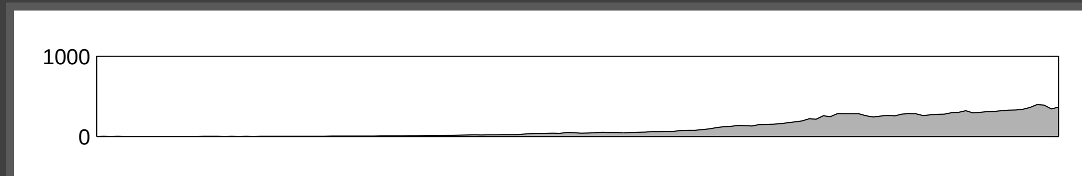
FLR(1)



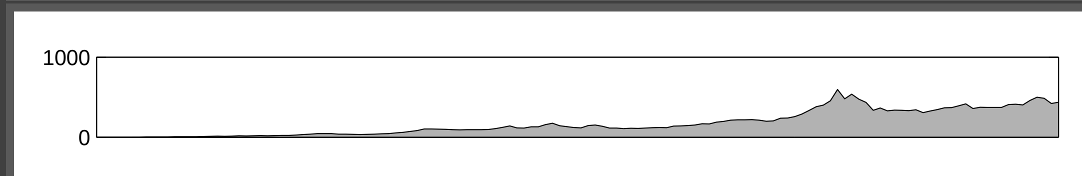
FLR(5)



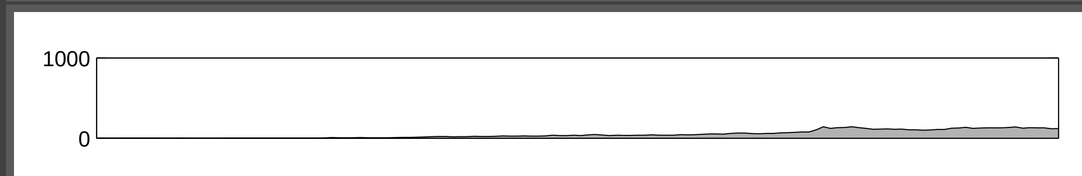
FLR(10)



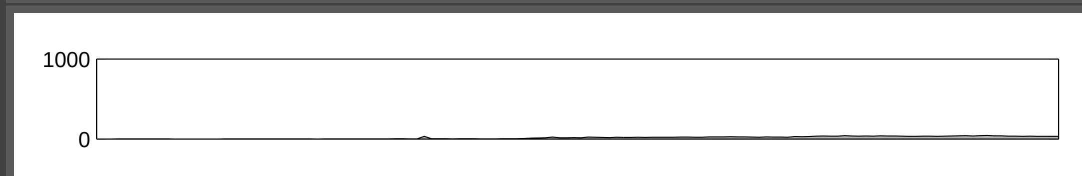
BP(1)



BP(5)



BP(10)



NEES ratio: NEES(alg) / NEES(FS2)

FS2

FLR(1)

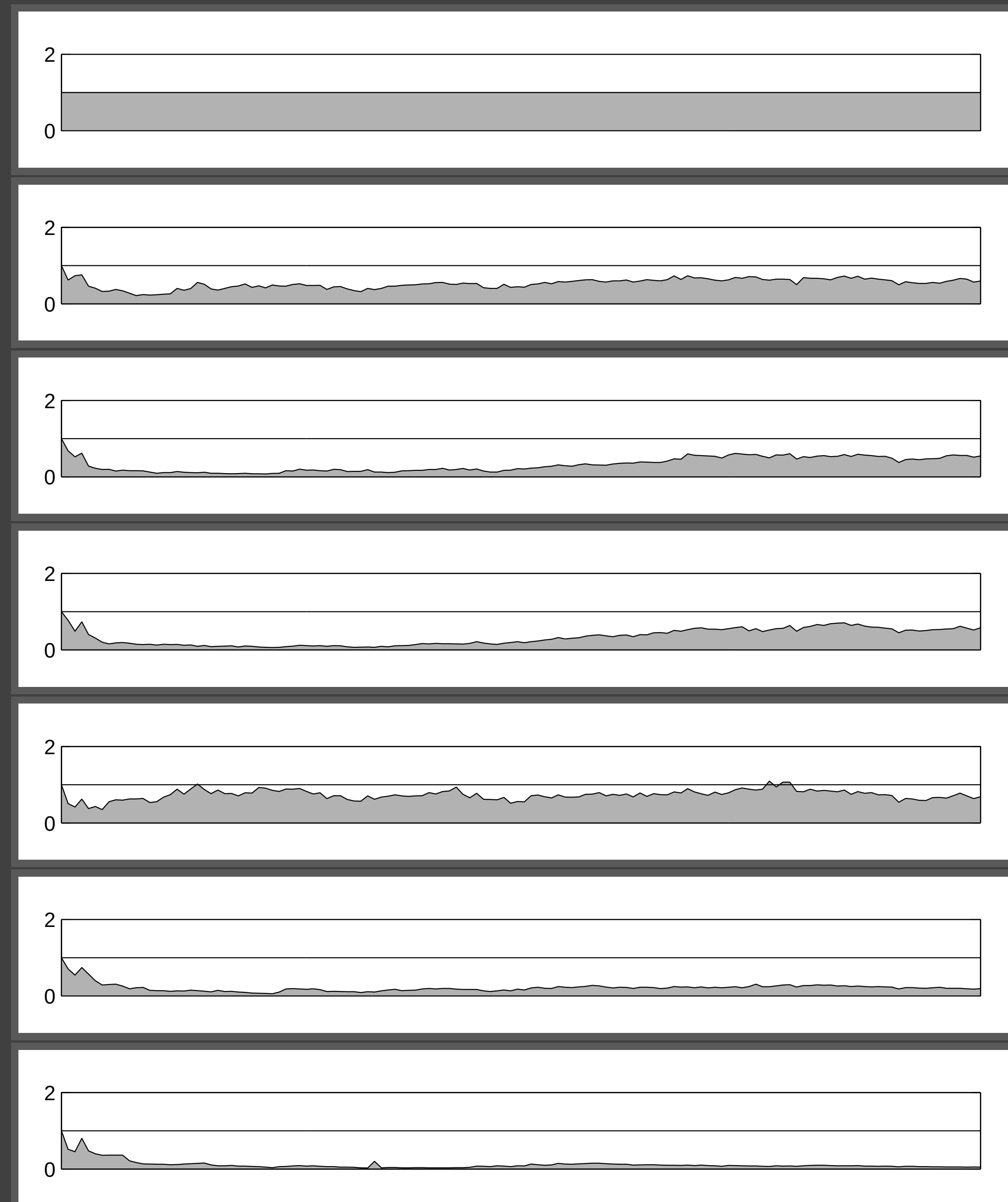
FLR(5)

FLR(10)

BP(1)

BP(5)

BP(10)



effective particles ($\hat{N}_{\text{eff}}$): $1/\sum_{i=1}^{N}\left(\omega_t^i\right)^2$

FS2

FLR(1)

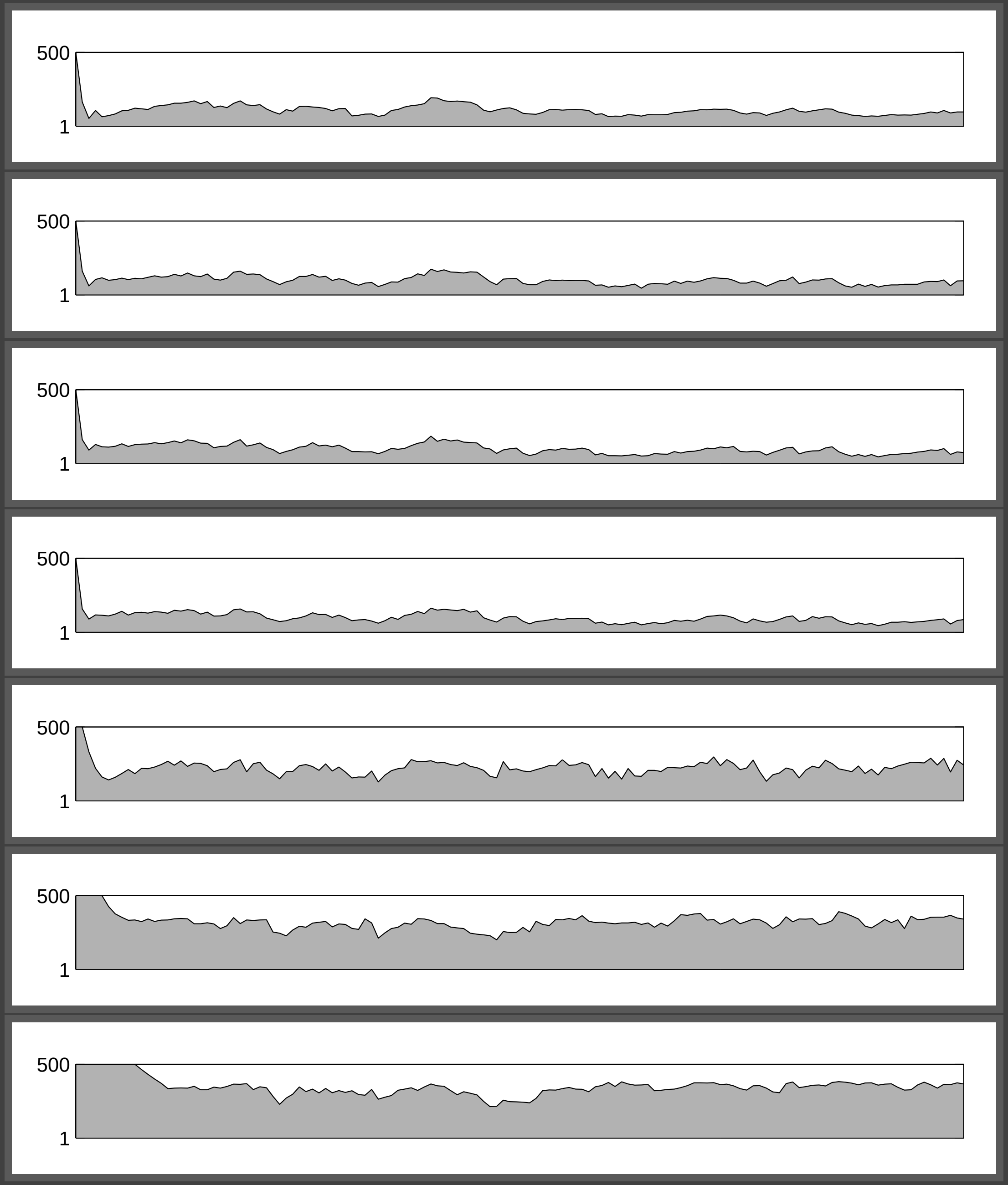
FLR(5)

FLR(10)

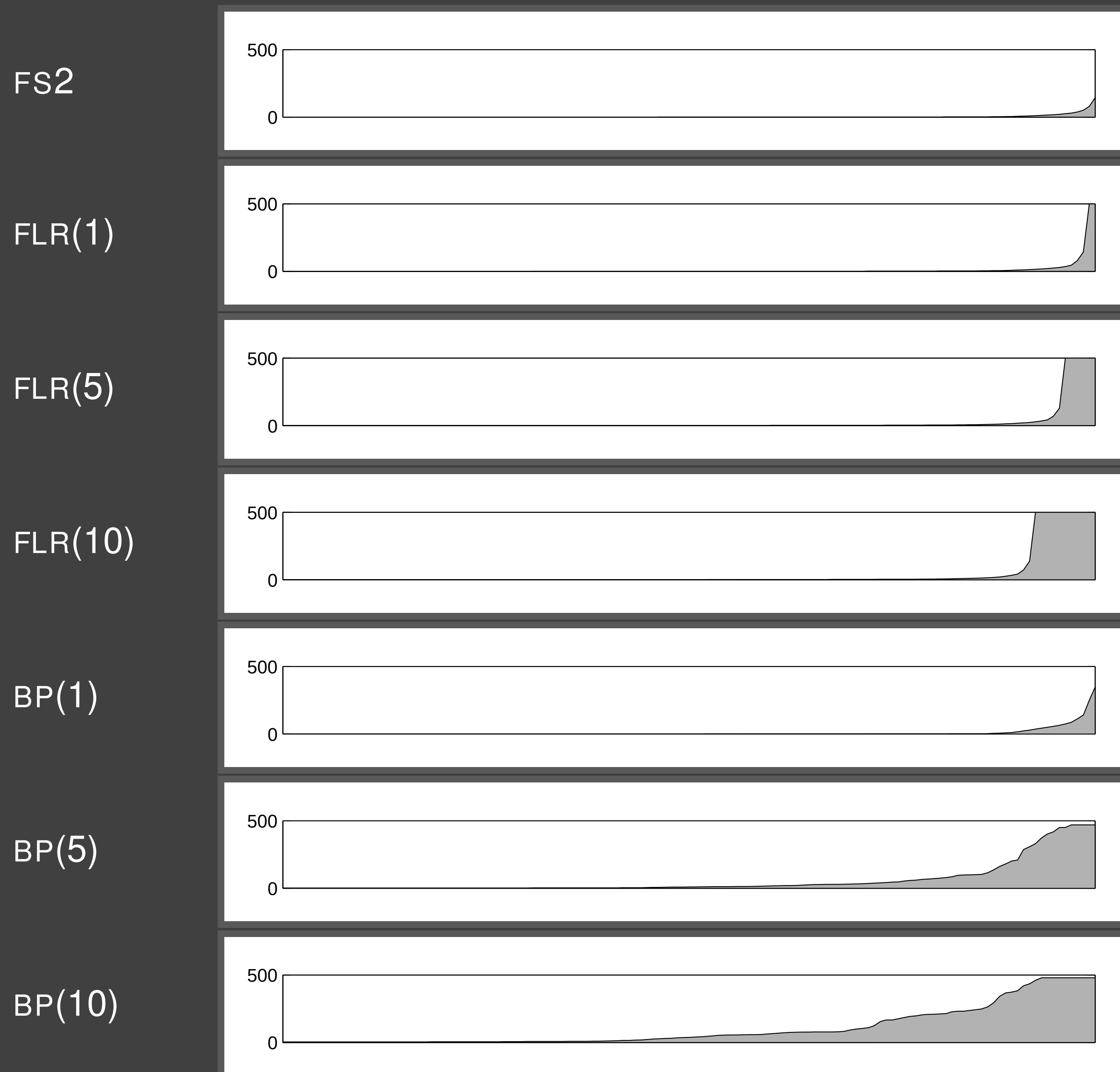
BP(1)

BP(5)

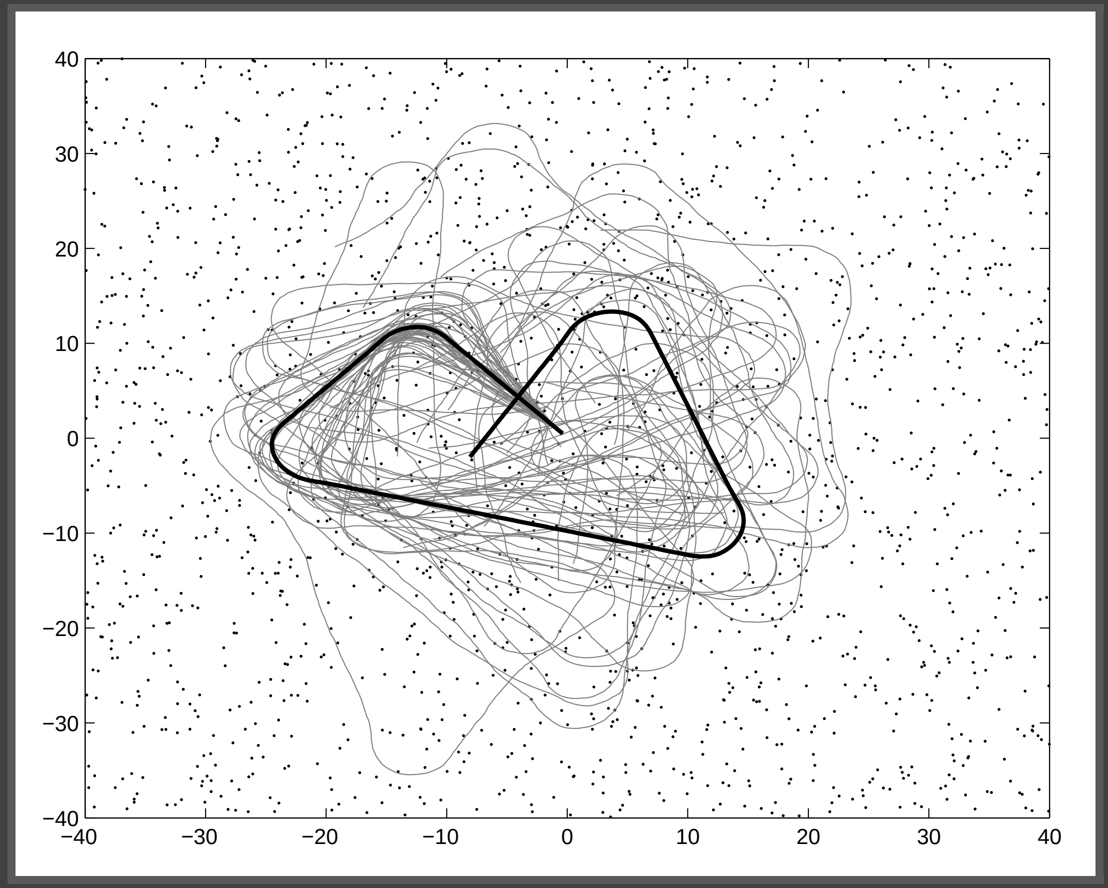
BP(10)



Unique samples of each pose: $|\{\mathbf{x}_k^{r,i} | \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t}\}|, k = 1 \ldots t$



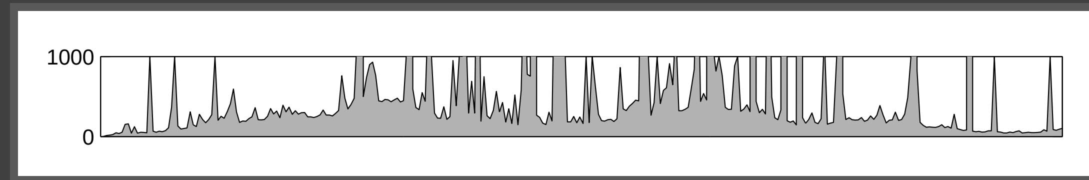
Results: dense environment



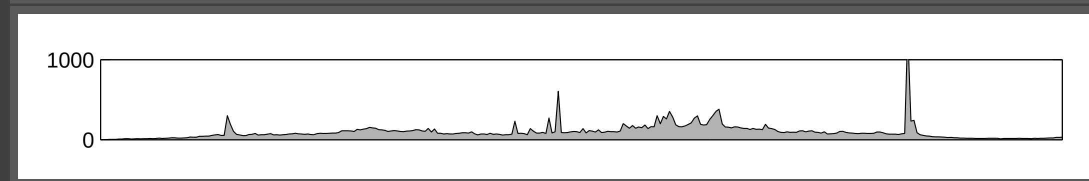
- 63 sec., loop
- 50 Monte Carlo trials averaged for all results

Norm. est. error sq. (NEES): $(\mathbf{x}_t^r - \hat{\mathbf{x}}_t^r)(\hat{\mathbf{P}}_t^r)^{-1}(\mathbf{x}_t^r - \hat{\mathbf{x}}_t^r)^T$

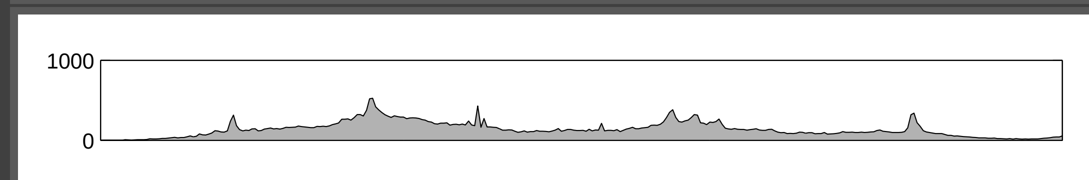
FS2



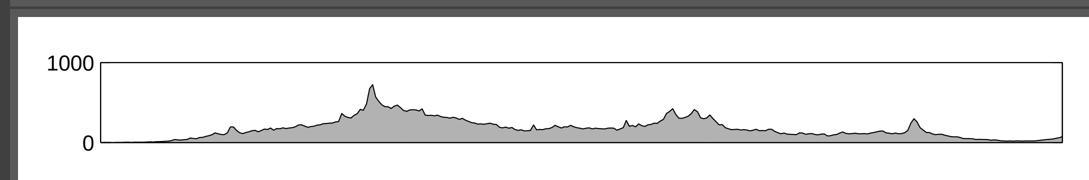
FLR(1)



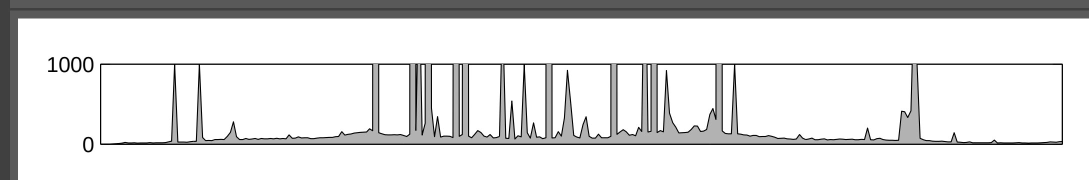
FLR(5)



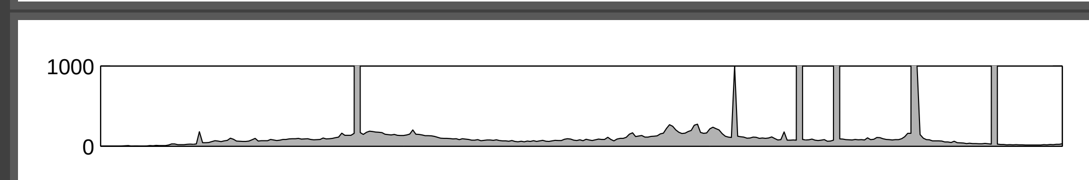
FLR(10)



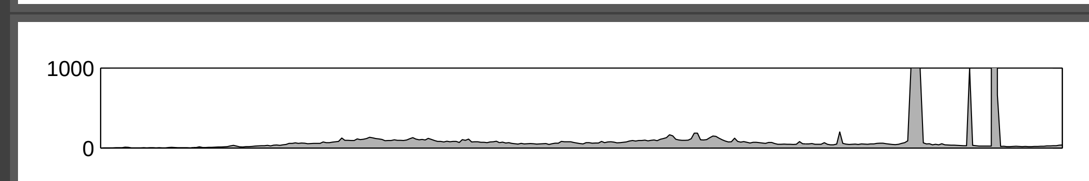
BP(1)



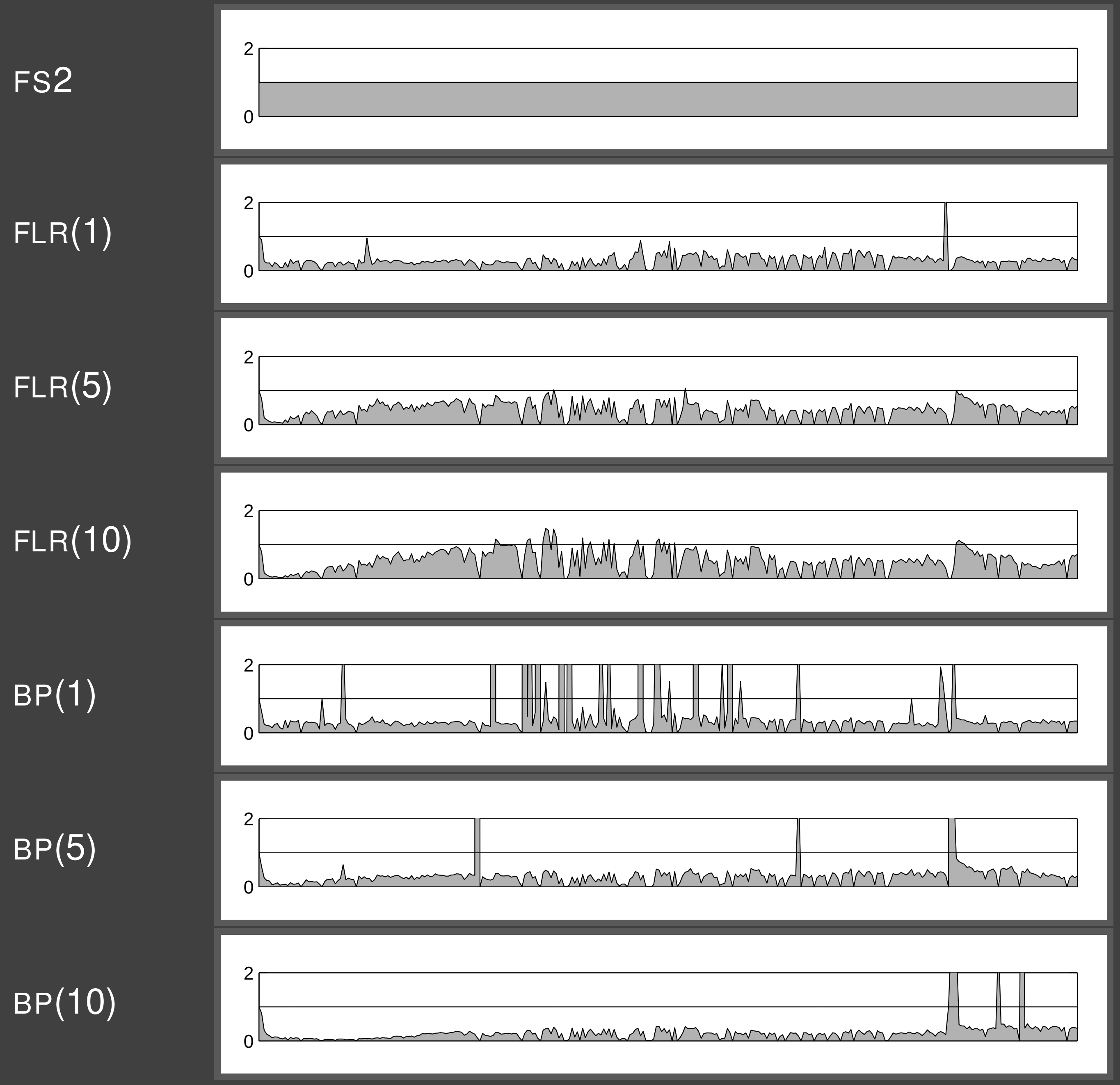
BP(5)



BP(10)



NEES ratio: NEES(alg) / NEES(FS2)



effective particles ($\hat{N}_{\mathbf{eff}}$): $1/\sum_{i=1}^{N}\left(\omega_t^i\right)^2$

FS2

FLR(1)

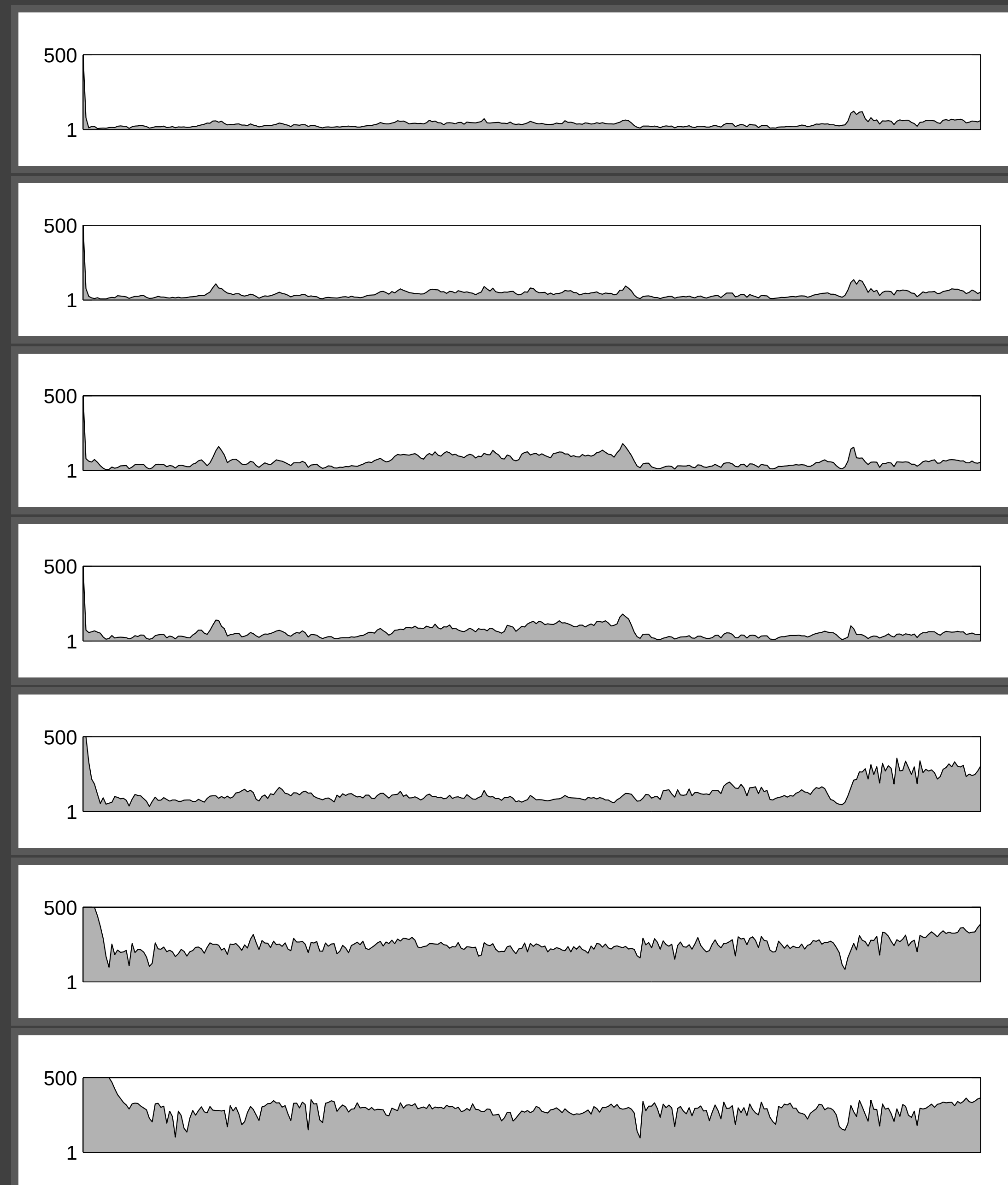
FLR(5)

FLR(10)

BP(1)

BP(5)

BP(10)



Unique samples of each pose: $|\{\mathbf{x}_k^{r,i}|\mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{n}_{1:t}\}|, k = 1 \dots t$

FS2

FLR(1)

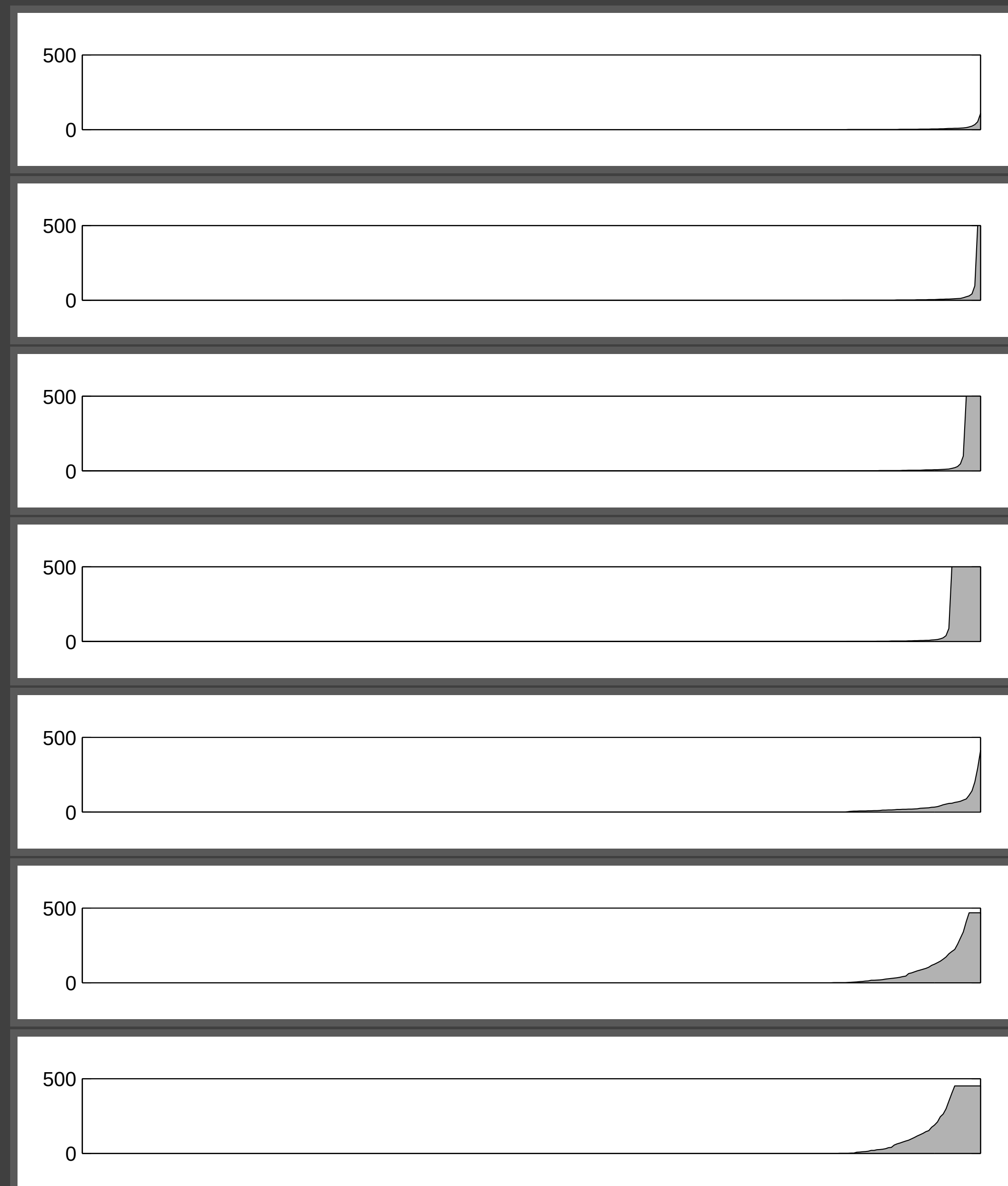
FLR(5)

FLR(10)

BP(1)

BP(5)

BP(10)



References

T. Bailey. *Mobile robot localisation and mapping in extensive outdoor environments*. PhD thesis, Australian Center for Field Robotics, University of Sydney, August 2002.

T. Bailey, J. Nieto, and E. Nebot. Consistency of the FastSLAM algorithm. In *IEEE Intl. Conf. on Robotics and Automation*, pages 424–427, 2006.

Y. Bar-Shalom, X. Rong Li, and T. Kirubarajan. *Estimation with applications to tracking and navigation*. Wiley, New York, 2001.

Kristopher R. Beevers. Sampling strategies for particle filtering SLAM. Technical Report 06-XX, Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, 2006.

Kristopher R. Beevers and Wesley H. Huang. Inferring and enforcing relative constraints in SLAM. In *Workshop on the Algorithmic Foundations of Robotics (WAFR 2006)*, New York, July 2006.

Kristopher R. Beevers and Wesley H. Huang. Fixed-lag sampling strategies for particle filtering SLAM. 2007 IEEE International Conference on Robotics and Automation (ICRA 2007), submitted, April 2007.

S. Chib. Calculating posterior distributions and modal estimates in Markov mixture models. *Journal of Econometrics*, 75(1):79–97, 1996.

A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo methods in practice*. Springer, New York, 2001.

A. Doucet, M. Briers, and S. Sénécal. Efficient block sampling strategies for sequential Monte Carlo methods. *Journal of Computational and Graphical Statistics*, to appear, 2006.

W.R. Gilks and C. Berzuini. Following a moving target — Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society B*, 63(1):127–146, 2001.

Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. Improving grid-based SLAM with Rao-Blackwellized particle filters by adaptive proposals and selective resampling. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation*, pages 2443–2448, 2005.

J.S. Liu and R. Chen. Blind deconvolution via sequential imputations. *Journal of the American Statistical Association*, 90(430):567–576, June 1995.

Jun S. Liu. *Monte Carlo strategies in scientific computing*. Springer, New York, 2001.

M. Montemerlo. *FastSLAM: a factored solution to the simultaneous localization and mapping problem with unknown data association*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 2003.

S.T. Pfister, S.I. Roumeliotis, and J.W. Burdick. Weighted line fitting algorithms for mobile robot map building and efficient data representation. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation*, Taipei, Taiwan, 2003.

Cyrill Stachniss, Giorgio Grisetti, and Wolfram Burgard. Recovering particle diversity in a Rao-Blackwellized particle filter for SLAM after actively closing loops. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation*, pages 667–672, 2005.