

Courant Institute of Mathematical Sciences
Mathematics Colloquium
November 8, 2010

Speaker: **Alexandre J. Chorin, UC Berkeley**

Title: Implicit Particle Filters for Data Assimilation

Particle filters for data assimilation are usually presented in the setting of stochastic ordinary differential equations (SODE). The task is to estimate the state of the solution of the SODE, conditioned by additional information provided by noisy observations of (possibly nonlinear) functions of the state. Particle filters approximate the resulting probability density functions by sequential Monte Carlo. At each step in time, one first guesses a "prior" incorporating the information contained in the SDE, which is then corrected by sampling weights determined by the observations, yielding a "posterior" density. The catch is that, in common weighting schemes, most of the weights become very small very fast, leaving only a small number of significantly weighted particles and sharply limiting the usefulness of the filters.

Implicit filters overcome this barrier by reversing the standard procedure. Rather than find samples and then determine their probabilities, implicit filters pick probabilities and then generate samples that assume them. The posterior density is written as $\exp(-F(x^{n+1}))$, where x^{n+1} is the next position of the particle (this defines a function F for each particle). One then represents x^{n+1} as a function of a fixed Gaussian random variable by solving the equation $F(x^{n+1}) - \min_{\xi} F = \xi^T \xi / 2$, where T denotes a transpose. The function F varies from particle to particle. The resulting x^{n+1} is a high probability sample of the posterior density with sampling weight $\exp(-\min F) J$, where J is the Jacobian of the map $\xi \rightarrow x^{n+1}$. There is a great deal of freedom in choosing the map $\xi \rightarrow x^{n+1}$ and this freedom can be used to produce maps which are easy to implement and whose Jacobian can be readily calculated.

I will demonstrate the usefulness of implicit filters by some numerical examples, and also extend the idea of implicit sampling to more general sampling problems.