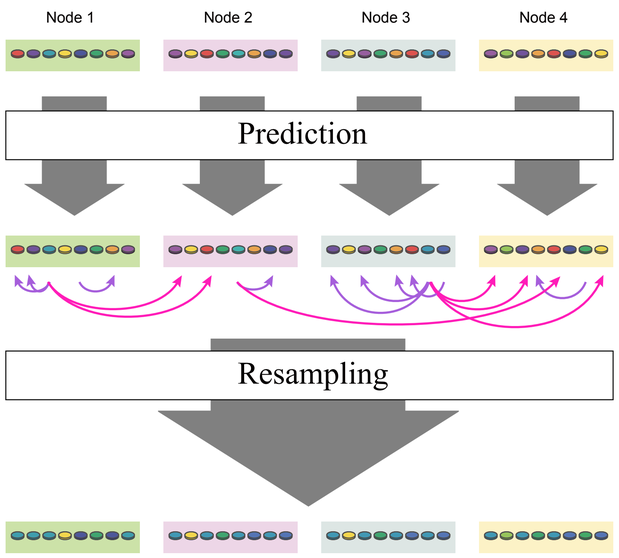
大規模並列計算機への粒子フィルタの実装に向けて

中野　慎也　モデリング研究系・助教

E-mail: shiny@ism.ac.jp

Motivation

Particle filter on a parallel computer

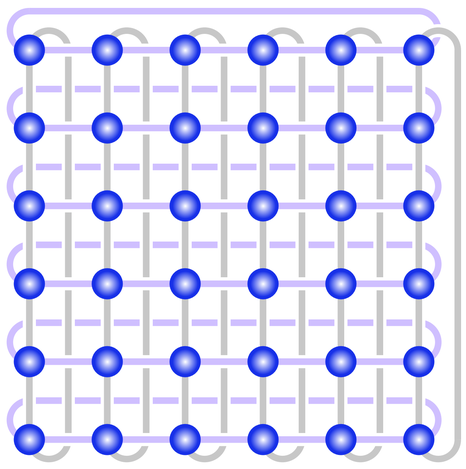


Prediction: easy to parallelize

Filtering: many data transfers occur randomly between different processing elements (PEs)

- The efficiency can be seriously affected by data transfers between different PEs in each filtering step.
- The resampling procedure in the filtering step is difficult to parallelize because it requires many random data transfers.

Parallel computer with a torus network



Two-dimensional torus network

The network topology of the Next-Generation Supercomputer in Japan, which is currently under development, is planned to be a virtually three-dimensional torus.

In a distributed computing system with a torus network, while a data transfer with an adjacent PE is fast, a data transfer with a distant PE is slow due to multiple hops and it might even impede data transfers between other PEs.

In the particle filter, inter-PE data transfers occur many times between randomly selected two PEs. Hence, data transfers between distant nodes may occur frequently.

Particle filter

The particle filter approximates a probability density function by a large number of particles. The predictive density function $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})$ at time t_k can be approximated as

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \approx \frac{1}{N}\sum_{i=1}^{N}\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{(i)}\right) \tag{1}$$

where $\{\boldsymbol{x}_{k-1|k-1}^{(1)}, \boldsymbol{x}_{k-1|k-1}^{(2)}, \cdots, \boldsymbol{x}_{k-1|k-1}^{(N)}\}$ is a set of samples drawn from $p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1})$, N is the number of the samples, and δ denotes Dirac's delta function.

Using the observation at time t_k, $\boldsymbol{y}_k$, a particle approximation of the filtered density function $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$ is obtaiend by using Bayes' theorem as follows:

$$\begin{aligned} p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) &= \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k)\,p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})}{\int p(\boldsymbol{y}_k|\boldsymbol{x}_k)\,p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})\,d\boldsymbol{x}_k} \\ &\approx \frac{1}{\sum_j p\left(\boldsymbol{y}_k|\boldsymbol{x}_{k|k-1}^{(j)}\right)}\sum_{i=1}^{N} p\left(\boldsymbol{y}_k|\boldsymbol{x}_{k|k-1}^{(i)}\right)\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{(i)}\right) \\ &= \sum_{i=1}^{N} w_i\,\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{(i)}\right) \end{aligned} \tag{2}$$

where $p(\boldsymbol{y}_k|\boldsymbol{x}_{k|k-1}^{(i)})$ is the likelihood of $\boldsymbol{x}_{k|k-1}^{(i)}$ given the data $\boldsymbol{y}_k$ and the weight w_i is defined as

$$w_i = \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_{k|k-1}^{(i)})}{\sum_j p(\boldsymbol{y}_k|\boldsymbol{x}_{k|k-1}^{(j)})}. \tag{3}$$

Then, we obtain a new ensemble $\{\boldsymbol{x}_{k|k}^{(1)}, \cdots, \boldsymbol{x}_{k|k}^{(N)}\}$ which approximates $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$ by resampling with replacement from the predictive ensemble $\{\boldsymbol{x}_{k|k-1}^{(1)}, \cdots, \boldsymbol{x}_{k|k-1}^{(N)}\}$ where $\boldsymbol{x}_{k|k}^{(i)}$ with a higher weight w_i is taken more frequently. The new ensemble may contain multiple copies of $\boldsymbol{x}_{k|k-1}^{(i)}$. which was originally contained in the predictive ensemble. Let the number of copies of $\boldsymbol{x}_{k|k-1}^{(i)}$ in the new ensemble, $n_k^{(i)}$, be an integer to satisfy

$$n_k^{(i)} \approx N w_i, \quad n_k^{(i)} \geq 0 \quad \text{(for all } i\text{)}, \quad \text{and} \quad \sum n_k^{(i)} = N. \tag{4}$$

Then, the filtered density function $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$ can be approximated by using the new ensemble as follows:

$$\begin{aligned} p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) &\approx \sum_{i=1}^{N} w_i\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{(i)}\right) \approx \sum_{i=1}^{N}\frac{n_k^{(i)}}{N}\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{(i)}\right) \\ &= \frac{1}{N}\sum_{i=1}^{N}\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k}^{(i)}\right). \end{aligned} \tag{5}$$

Grouping

Define $\Omega_k^{(\mu)}$ and $\omega_k^{(\mu,\nu)}$ such that

$$\Omega_k^{(\mu)}\omega_k^{(\mu,\nu)} = w_k^{([\mu-1]\lambda+\nu)} \qquad \sum_{\nu=1}^{\lambda}\omega_k^{(\mu,\nu)} = 1 \qquad \sum_{\mu=1}^{\Lambda}\Omega_k^{(\mu)} = 1 \qquad \Lambda\lambda = N.$$

Then,

$$\begin{aligned} p(\boldsymbol{x}_k|\boldsymbol{y}_{1:t}) &\approx \sum_{i=1}^{N} w_k^{(i)}\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{(i)}\right) \\ &= \sum_{\mu=1}^{\Lambda}\Omega_k^{(\mu)}\sum_{\nu=1}^{\lambda}\omega_k^{(\mu,\nu)}\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{([\mu-1]\lambda+\nu)}\right). \end{aligned} \tag{6}$$

Similarly to Eq. (5), we can approximate each term of the summation in Eq. (6) as

$$\Omega_k^{(\mu)}\sum_{\nu=1}^{\lambda}\omega_k^{(\mu,\nu)}\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{([\mu-1]\lambda+\nu)}\right) \approx \Omega_k^{(\mu)}\sum_{\nu=1}^{\lambda}\frac{m_k^{(\mu,\nu)}}{\lambda}\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{([\mu-1]\lambda+\nu)}\right) \tag{7}$$

where $m_k^{(\mu,\nu)}$ satisfies

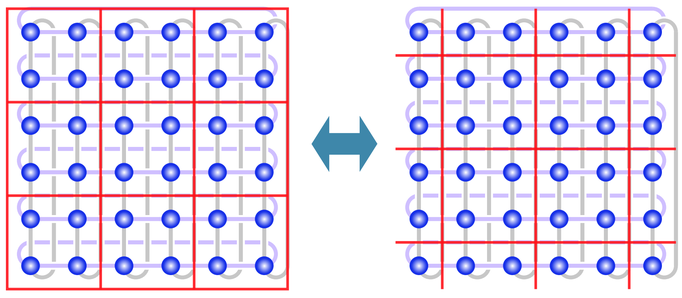
$$m_k^{(\mu,\nu)} \approx \lambda\omega_k^{(\mu,\nu)}, \quad m_k^{(\mu,\nu)} \geq 0, \quad \text{(for all } \mu \text{ and } \nu\text{)}, \quad \text{and} \quad \sum_{\nu=1}^{\lambda} m_k^{(\mu,\nu)} = \lambda \quad \text{(for all } \mu\text{)}. \tag{8}$$

We devide the predictive ensemble $\{\boldsymbol{x}_{k|k-1}^{(i)}\}_{i=1}^{N}$ into Λ subsets, each of which contains λ particles, and resample with replacement from each subset $\{\boldsymbol{x}_{k|k-1}^{([\mu-1]\lambda+\nu)}\}_{\nu=1}^{\lambda}$ to obtain a set of particles which contains $n_k^{([\mu-1]\lambda+\nu)}$ copies of $\boldsymbol{x}_{k|k-1}^{([\mu-1]\lambda+\nu)}$ for each μ. Using the new subset $\{\boldsymbol{x}'^{([\mu-1]\lambda+\nu)}_{k|k}\}_{\nu=1}^{\lambda}$, we can approximate the filtered distribution as

$$\begin{aligned} p(\boldsymbol{x}_k|\boldsymbol{y}_{1:t}) &\approx \sum_{\mu=1}^{\Lambda}\Omega_k^{(\mu)}\sum_{\nu=1}^{\lambda}\frac{m_k^{(\mu,\nu)}}{\lambda}\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}^{([\mu-1]\lambda+\nu)}\right) \\ &= \sum_{\mu=1}^{\Lambda}\frac{\Omega_k^{(\mu)}}{\lambda}\sum_{\nu=1}^{\lambda}\delta\left(\boldsymbol{x}_k - \boldsymbol{x}_{k|k}^{([\mu-1]\lambda+\nu)}\right). \end{aligned} \tag{9}$$

Resampling within a group

We can divide the particles in the ensemble into multiple groups, and we can resample the particles within each group. Although each group offers an approximation of the filtered probability density function (PDF), the integration of all the group reinforce the estimation of the filtered PDF.



- We divide the PEs in use into multiple groups. Each of particles is divided into one of the groups according to the PE which the particle is assigned to.
- We may resample the particles within each PE group.
- We can even change the grouping at each time step; that is, PEs which were previously embraced by different groups can be put together to form a new group (Bolić et al. 2005*).
- We consider a strategy to use two grouping patterns which is suitable to a torus network topology. In this strategy, the two grouping patterns are alternately switched.
- By dynamically switching between the two grouping patterns, each node is put together with three nodes each of which was embraced by a different group. The information of a particle with high likelihood may propagate to all the nodes after several time steps.

* Bolić et al.: 'Resampling algorithms and architectures for distributed particle filters. *IEEE Trans. Signal Processing*, v. 53, p. 2442, 2005.

Experiments using Lorenz 96 model

- We performed experiments using the Lorenz 96 model (Lorenz and Emanuel, 1998), in which the evolution of each component x_j is written as

$$\frac{dx_j}{dt} = (x_{j+1} - x_{j-2})x_{j-1} - x_j + 8$$

where $j = 1, \ldots, J$, $x_{-1} = x_{J-1}$, $x_0 = x_J$, and $x_{J+1} = x_1$. Here, the dimension J was set as 40. One time step was set as 0.005.

We generated artificial data from a run with a certain initial condition. The artificial data are generated

- every 10 time steps
- with artificial noises having a standard deviation of 1.5.

It was also assumed that the data for x_j can be acquired if j is an even number ($j = 2, \ldots, 40$); that is, half of the state variables are observed.

- In assimilating the artificial data, the system noise was assumed to be a Gaussian noise with zero mean and a diagonal covariance as diag(0.25, …, 0.25). The likelihood was calculated as follows:

$$p(\boldsymbol{y}_k|\boldsymbol{x}_{k|k-1}) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left[-\frac{\|\boldsymbol{y}_k - H\boldsymbol{x}_{k|k-1}\|^2}{2\sigma^2}\right]$$

where $\boldsymbol{y}_k$ is the observation vector $(y_{1,k}, \ldots, y_{20,k})$ and σ was specified as 3. The operator H extracts the observable components from the state vector $\boldsymbol{x}_{k|k-1}$; that is, $H\boldsymbol{x}_{k|k-1} = (x_{2,k|k-1}\, x_{4,k|k-1} \ldots x_{40,k|k-1})^T$.

Result using global particle filter

PEs	Particles	RMS dev.	Elapsed time
4 × 4	65536	0.85	01m12s
8 × 8	262144	0.77	03m55s
16 × 16	1048576	0.73	17m18s

Result using the alternate switching method

PEs	Particles	RMS dev.	Elapsed time
4 × 4	65536	0.84	00m40s
8 × 8	262144	0.77	00m54s
16 × 16	1048576	0.73	01m45s

RMS dev.: The root-mean-squares between the estimates and the true values over 10000 steps

The elapsed times were measured on a super-computer of ISM (ismrx). Here, each core is regarded as one PE, and each PE is regarded as one virtual node of the two-dimensional torus network.

Summary

- The present study proposes a dynamic grouping strategy which achieves high efficiency on a massively parallel computing system.
- In the proposed strategy, the resampling at each time step is performed within each PE group. Data transfers are thus restricted within the PE group.
- Since the resampling for different PE groups can be done in parallel, the time cost would be remarkably reduced.
- The grouping is not static but the two grouping patterns are alternately switched. This switching allows us to maintain the diversity of the ensemble because particles assigned to each node are exchanged with various other nodes by changing the grouping.
- It has been confirmed that this strategy is notably effective especially for the cases with a large number of nodes in which the normal global PF is prohibitive.