Frontiers of Information Technology & Electronic Engineering
www.zju.edu.cn/jzus; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Resampling methods for particle filtering: identical distribution, a new method, and comparable study*

Tian-cheng LI†[1,2], Gabriel VILLARRUBIA[1], Shu-dong SUN[2], Juan M. CORCHADO[1,4], Javier BAJO[3]

([1]BISITE Group, Faculty of Science, University of Salamanca, C/Espejo s/n, Salamanca 37008, Spain)
([2]School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an 710072, China)
([3]Department of Artificial Intelligence, Technical University of Madrid, Calle Ramiro de Maeztu 7, Madrid 28040, Spain)
([4]Osaka Institute of Technology, Asahi-ku Ohmiya, Osaka 535-8585, Japan)
†E-mail: t.c.li@usal.es; t.c.li@mail.nwpu.edu.cn

Received June 24, 2015; Revision accepted Sept. 1, 2015; Crosschecked Sept. 10, 2015; Online published Oct. 26, 2015

Abstract: Resampling is a critical procedure that is of both theoretical and practical significance for efficient implementation of the particle filter. To gain an insight of the resampling process and the filter, this paper contributes in three further respects as a sequel to the tutorial (Li *et al.*, 2015). First, identical distribution (ID) is established as a general principle for the resampling design, which requires the distribution of particles before and after resampling to be statistically identical. Three consistent metrics including the (symmetrical) Kullback-Leibler divergence, Kolmogorov-Smirnov statistic, and the sampling variance are introduced for assessment of the ID attribute of resampling, and a corresponding, qualitative ID analysis of representative resampling methods is given. Second, a novel resampling scheme that obtains the optimal ID attribute in the sense of minimum sampling variance is proposed. Third, more than a dozen typical resampling methods are compared via simulations in terms of sample size variation, sampling variance, computing speed, and estimation accuracy. These form a more comprehensive understanding of the algorithm, providing solid guidelines for either selection of existing resampling methods or new implementations.

Key words: Particle filter, Resampling, Kullback-Leibler divergence, Kolmogorov-Smirnov statistic
doi:10.1631/FITEE.1500199 **Document code:** A **CLC number:** TN713

1 Introduction

Dynamic state estimation, namely filtering, is concerned with the sequential process of estimating a state that evolves over time and that is periodically observed. The Bayesian filter forms the standard solution to infer the state from noisy observations, which requires the complete posterior density of the state to be determined as a function of time. The posterior probability density function (PDF) can be analytically computed for systems only with linear dynamics and additive Gaussian noises, for which the known Kalman filter gives the optimal estimate (Kalman, 1960). In the general case of nonlinear systems and/or non-Gaussian noises, it is impossible to compute the exact form of the posterior PDF. Simply, a Gaussian variable after nonlinear transformation or affected by non-Gaussian noises will be no more Gaussian; instead, one has to resort to approximations. Approximations can be parametric or nonparametric. In the former case, the posterior PDF is represented by a family of functions that are fully characterized by the parameters. For example, the first and second statistical moments (namely, mean and variance) are sufficient to determine a Gaussian distribution. However, they are not sufficient to represent a general probability distribution. In the latter

* Project supported by the National Natural Science Foundation of China (No. 51475383), European Commission MSCA-RISE-2014 (No. 641794), the Excellent Doctorate Foundation of Northwestern Polytechnical University, and the Postdoctoral Fellowship of the University of Salamanca
ORCID: Tian-cheng LI, http://orcid.org/0000-0002-0499-5135

case where the posterior PDF can be an arbitrary distribution (whether non-Gaussian or multimodal), parameterization is impossible or will suffer from significant approximation errors. In these cases, the posterior PDF must be approximated with a nonparametric PDF. Among them, the particle filter (PF), also referred to as the sequential Monte Carlo (SMC) approach, has gained the most popularity (Doucet *et al.*, 2001; Arulampalam *et al.*, 2002; Djurić *et al.*, 2003; Cappé *et al.*, 2007; Gustafsson, 2010; Li *et al.*, 2014), especially for strong nonlinear filtering models that may be affected by non-Gaussian noises.

The PF drives the Bayesian recursions over time using a set of weighted particles, where the random measure formed is theoretically qualified to represent any filtering distribution. The filter evaluates how well each particle conforms to the dynamic model and explains the observations, using this assessment to update the particle set online and hence form Bayesian posterior estimates. The three key operations forming the basis of the most commonly used type of the PF called the sampling importance resampling (SIR) or sequential importance sampling and resampling (SISR) filter (Rubin, 1987; Gordon *et al.*, 1993) are:

1. Particle propagation: updating the state of particles according to the state transition process or an alternative proposal;

2. Weight computation: updating the weight of particles that are probability masses associated with the particles with reference to the observation;

3. Resampling: sampling a new set of particles from the original particle population for better particle representation of the filtering distribution.

The former two steps comprise the sequential importance sampling (SIS) paradigm. Resampling is an essential component for the PF because without it, the SIS will quickly produce a degenerate set of particles (Gordon *et al.*, 1993; Kong *et al.*, 1994; Arulampalam *et al.*, 2002; Pérez *et al.*, 2005; Li *et al.*, 2014) as the weight discrepancy between particles increases with time. This means that, with very few exceptions, most particles will have negligible weights and, as a result, the random measure they form is unreliable. To combat this, resampling replaces the particle set by a new set of equivalently or similarly weighted particles, resembling the pruning and enrichment operation (Crisan *et al.*, 1998; Chen *et al.*, 2005). Our recent tutorial (Li *et al.*, 2015) (hereafter referred to as 'the tutorial') has provided a comprehensive overview and classification of state-of-the-art resampling methods. To serve as a further extension and supplement to the tutorial, this paper puts forward identical distribution (ID) as a basic principle for resampling and correspondingly proposes a new resampling algorithm that has the best ID and compares representative resampling methods. In addition, more works on resampling will be reviewed with a minimum overlap to the works that have been introduced in the tutorial. By combining the new findings with the tutorial, a solid guideline for designing new resampling methods is formed while a more comprehensive understanding of the resampling is expected.

It is necessary to note that resampling is a broadly-used concept involved in many statistical problems such as data analysis (Efron *et al.*, 2015), exhibiting very different contents therein. The first standard PF is referred to as the bootstrap filter (Gordon *et al.*, 1993), where the bootstrapping that relies on random sampling with replacement falls just in the broader class of resampling methods. It is fair to say, algorithms and applications of resampling are general, but this paper concentrates only on its implementation and application within particle filtering to avoid over-wide discussion. Since this is a sequel to the tutorial, we have no intention of repeating materials; the reader is referred to the tutorial for the details of existing resampling methods.

2 Why resampling?

Since resampling is a relatively independent procedure that is in general not correlated with the other operations required by the PF, we focus on the resampling operation only while omitting the other filtering operations for which there exist a variety of different versions. This allows a clear understanding of our contributions. We start basically with the SMC approximation of the Bayesian posterior distribution of the state

$$p(x_t \mid y_{1:t}) \approx \sum_{m=1}^{M} w_t^{(m)} \delta(x_t - x_t^{(m)}), \tag{1}$$

where t is the time, $x_t \in \mathbb{R}^{d_x}$ the state, $y_t \in \mathbb{R}^{d_y}$ the observation, $x_t^{(m)}$ and $w_t^{(m)}$ the state and weight of the *m*th particle respectively, *M* the total number of particles, and $\delta(\cdot)$ the Dirac delta impulse.

Resampling was initially employed to combat sample degeneracy (Gordon *et al.*, 1993; Kong *et al.*, 1994) by sampling a new (equivalently or similarly weighted) particle set $\tilde{\chi}_t = \{\tilde{x}_t^{(n)}, \tilde{w}_t^{(n)}\}_{n=1}^{N}$ from the degenerate set $\chi_t = \{x_t^{(m)}, w_t^{(m)}\}_{m=1}^{M}$ for replacement at time *t*, where $\tilde{x}_t^{(n)}$ and $\tilde{w}_t^{(n)}$ give the state and weight of a resampled particle, respectively, and *N* is the number of particles obtained from resampling (which is specified by the user and is often equal to *M*). There are different aspects to understand this process. In what follows, we investigate from the perspective of tracking loss.

Note that if a sample is too far away from the true state, we say it has lost the estimate; i.e., under a certain confidence level, we believe that the particle is lost. To measure this, we define the tracking loss probability function $R(x_t - x_t^{(m)})$ that can be conveniently interpreted as a risk function of the loss of a particle, which gives the probability that particle $x_t^{(m)}$ has lost the true state x_t as follows:

$$R(x_t - x_t^{(m)}) := \alpha \, | x_t - x_t^{(m)} |^{\beta}, \tag{2}$$

where $|\cdot|$ is a distance defined in the state space and α, β are nonnegative coefficients. This probability can be interpreted to mean that the farther a sample is from the true state, the higher the probability that it loses the estimate is. On the other hand, the likelihood principle indicates that the farther a sample is from the true state, the lower its likelihood is and the smaller its weight $w_t^{(n)}$ is; i.e.,

$$R(x_t - x_t^{(m)}) \propto | x_t - x_t^{(m)} | \propto (w_t^{(m)})^{-1}. \tag{3}$$

Following these two lines of thinking, the estimate is lost by all the particles with probability

$$R(\chi_t) := \prod_{m=1}^{M} R(x_t - x_t^{(m)}) \propto (\prod_{m=1}^{M} w_t^{(m)})^{-1}. \tag{4}$$

This can be taken as the probability that the filter (or the random measure χ_t) loses tracking.

Resampling, generally replicating high-weighted particles to replace low-weighted particles, is essentially reducing the probability that the filter loses tracking, obtaining $R(\chi_t) \geq R(\tilde{\chi}_t)$. However, this comes at the price of discarding low-weighted particles (except few special implementations that advocate preserving low-weighted particles through special strategies (Godsill *et al.*, 2007; Li *et al.*, 2012; Choe *et al.*, 2014)), which are generally located in the tail of the distribution, and are helpful in dealing with outlier/disturbances, potentially leading to sample impoverishment. That is, most of the particles are duplicated from the same few ancestors, suffering from a problem of the same nature as degeneracy. The trade-off between degeneracy and impoverishment, and their treatments have been reviewed in Li *et al.* (2014), while the side effects of resampling and corresponding compensative strategies have been discussed in the tutorial. We reiterate that there is a trade-off between robustness and accuracy during filtering. The former requires maintaining good diversity of the particle population (to form a relatively heavy tail of the filtering distribution) to deal with outliers, while the latter requires the particle population to be concentrated. More in-depth results on convergence (Crisan and Doucet, 2002; Hu *et al.*, 2011; Stano *et al.*, 2013; Mbalawata and Särkkä, 2016), stability (Whiteley, 2013; Beskos *et al.*, 2014; Douc *et al.*, 2014), and concentration (Del Moral *et al.*, 2012) of the PF can be found.

The motivation of resampling includes, but is not limited to, combating sample degeneracy (while avoiding impoverishment) and adjusting the number of particles online (Li *et al.*, 2013b; 2016). The achievement of resampling tends to be less restricted to a specified form but indeed provides much freedom for the user to improve the quality of the particle set from different aspects. In particular, four special classes of resampling strategies are worth noting here and may inspire further developments.

1. Instead of sampling from the original particle set χ_t, resampling may be applied on an alternative of the original measure χ_t as addressed in the tutorial about resampling (Liu *et al.*, 2001), and there are many other alternatives (Kwak *et al.*, 2008; Choe *et al.*, 2015). These resampling implementations are almost surely biased.

2. Instead of sampling in the one-dimensional weight space only, there are a few exceptions that take into consideration the state of particles for resampling. In spite of the computational complexity, this seems to be particularly helpful in the context of robot Monte Carlo localization (Li and Sun, 2010; Adiprawita *et al.*, 2011; Morelande and Zhang, 2011; Li *et al.*, 2012).

3. Instead of resampling at most once during each filtering iteration, the frequency of resampling can be higher. For example, a double-resampling strategy was used by Li and Sun (2010) and Lang *et al.* (2015), where resampling was implemented twice per filtering iteration. One is before the weight-updating step, which is a particle-merging process to reduce the number of particles that need weight updating, and the other is after the weight-updating step, with the goal to restore/increase the number of particles involved in the state-prediction step. As a result, a good trade-off between filtering accuracy (the more particles for state prediction, the better) and computational efficiency (the fewer particles for weight updating, the better) can be achieved.

4. In the case of a time-varying number of targets, e.g., the SMC implementation of the probability hypothesis density filter (Li *et al.*, 2016), where targets may appear or disappear randomly both in the state space and in time, resampling is responsible for the control of the number of particles: increasing it when new targets appear and reducing it when existing targets disappear. Here, what convenient for resampling parallelization is that particle weights do not need to be normalized, which is different from the implementation in the general PF.

Furthermore, hybrid methods that integrate different sampling strategies (Zhi *et al.*, 2014) can benefit in multiple aspects. Despite these special implementations, resampling must still follow certain rules in the general sense, one of which is that the random measure formed by the weighted particles shall not be changed significantly because of resampling, unless it is required to do so with a reason—we state this because there are cases in which one may tend to change the distribution for one reason or another, e.g., the former two classes of special implementations mentioned above and the study by Das and Mazumdar (2013). We will focus on this in the following section.

3 Identical distribution for resampling

The strength of the nonparametric PF over parametric filters is the ability to approximate the complete distribution of interest, which can be arbitrary. To guarantee this strength, the resampled particle system shall be as good an approximation to the original distribution as possible, in some suitable sense (Douc and Cappé, 2005). First, resampling should be unbiased so that it will not ‘drift’ the estimate (which is often given by the mean of the distribution). Apart from the unbiasedness, greater attention should be paid to a higher-level measure of the change of the underlying posterior distribution caused by resampling.

We argue that as long as no new observation is used in the process of resampling, the distribution of particles before and after resampling should be identical; namely, identical distribution (ID) should be followed. However, except in very rare cases, the distributions before and after resampling are more or less different. Therefore, it is necessary both in theory and in practice to know how much change the resampling process makes to the distribution or, rather, how well the resampling is able to preserve the original distribution. Three metrics are introduced below in order to assess the ID attribute of different resampling methods.

3.1 Kullback-Leibler divergence

The Kullback-Leibler (KL) divergence (KLD) (Kullback and Leibler, 1951), also known as information divergence or relative entropy, is a widely used asymmetric measure of the difference between two probability distributions. Specifically, the KLD of q from p, denoted as $D_{\mathrm{KL}}(p||q)$, is a measure of the information lost when q is used to approximate p. For the discrete probability distributions p and q, $D_{\mathrm{KL}}(p||q)$ is given by

$$D_{\mathrm{KL}}(p\,||\,q)=\sum_{i}p(i)\ln\frac{p(i)}{q(i)}. \tag{5}$$

Given that p is the probability distribution of the particles before resampling and q is that after resampling, Eq. (5) provides a measure of the information loss caused by resampling, which can be interpreted as an index of the ID attribute of the resampling process. Obviously, a smaller KLD (less information loss) indicates a better ID. However, the KLD is not really a metric since it is not symmetric (Topsoe, 2000), but it can be easily extended to be a symmetry-enhanced KL-divergence (S-KLD), e.g.,

$$\begin{aligned} D_{\text{S-KL}}(p \| q) &:= \frac{D_{\text{KL}}(p \| q) + D_{\text{KL}}(q \| p)}{2} \\ &= \frac{1}{2}\sum_i (p(i) - q(i)) \ln \frac{p(i)}{q(i)}. \end{aligned} \tag{6}$$

Eq. (6) in fact gives the average of the cross-entropies minus the average of the entropies. One of the drawbacks of the KLD and S-KLD is that they may be unbounded. For more symmetric divergences inspired by KLD and related models such as the Jensen-Shannon divergence and Bhattacharyya distance, please refer to Nielsen (2010).

Arguably, a larger number of particles (namely, smaller KLD of the resampled particle set from the original particle set) is helpful in reducing information loss, but will also increase computation. This is another trade-off between the computing efficiency and ID attribute for resampling. Therefore, a principle that determines the minimum number of particles to resample conditioned on a sufficient ID quality is that the KLD of the resampled particle set from the original set does not exceed a prespecified error bound; i.e., the information lost because of resampling needs to be bounded. The readers are referred to Fox (2003) and Li *et al.* (2013b) for details. The KLD has also been used for distributed filtering assessment (Wang and Djurić, 2013). In addition to the number of particles to be resampled, another critical factor affecting the ID quality of the resampling is the sampling procedure used.

3.2 Kolmogorov-Smirnov statistic

The Kolmogorov-Smirnov (K-S) statistic, also referred to as *F*-discrepancy, is the maximum vertical distance between two empirical distribution functions (EDF):

$$D_{\text{K-S}} = \sup_{-\infty<x<\infty} |F_{1,M}(x) - F_{2,N}(x)|, \tag{7}$$

where $F_{1,M}(\cdot)$ and $F_{2,N}(\cdot)$ are the EDFs of two distributions, respectively. For our case, the discrete EDF F_N for N independent samples $x_t^{(n)}$ (that is weighted $w_t^{(n)}$) is defined as follows:

$$F_N(x) = \sum_{n=1}^{N} w_t^{(n)} I_{x^{(n)} \le x}, \tag{8}$$

where $I_{x^{(n)} \le x}$ is the indicator function. If $x^{(n)} \le x$, it is equal to 1; otherwise, it is equal to 0.

The K-S statistic aims to identify and capture the largest discrepancy between two empirical distributions. It has been used for model assessment (Djurić and Miguez, 2010) and for measuring the difference between two distributions before and after resampling (Li and Sun, 2010; Li *et al.*, 2012). Obviously, the smaller the *F*-discrepancy, the better the ID quality of resampling. We investigate the largest possible *F*-discrepancy caused by several typical unbiased single-resampling methods, including multinomial resampling (MR) (Gordon *et al.*, 1993), residual resampling (RR) (Liu and Chen, 1998), stratified resampling (StR), systematic resampling (SyR) (Kitagawa, 1996), residual systematic resampling (RSR) (Bolić *et al.*, 2003), branch-kill resampling (B-k) (Crisan and Lyons, 1999), and rounding-copy resampling (R-c) (Li *et al.*, 2013a). The *F*-discrepancy analysis is given in Appendix A (assuming that the same number of particles is resampled in all resampling methods), which concludes that

$$D_{\text{MR}} > D_{\text{StR}} \ge D_{\text{SyR}} \cong D_{\text{RSR}} \cong D_{\text{B-k}} \cong D_{\text{R-c}}. \tag{9}$$

Furthermore, the deterministic resampling (Li *et al.*, 2012) that avoids discarding particles can achieve the lowest *F*-discrepancy, given that the state space is properly partitioned. However, the number of particles it outputs is not constant. Relevantly, the quasi-Monte Carlo-based SIR is verified to be empirically better than the original SIR (Pérez *et al.*, 2005) for obtaining smaller *F*-discrepancy. More strategies to obtain smaller sampling discrepancy, or to say better ID performance, can be found in Crisan *et al.* (1998) and Robert and Casella (1999).

Generally, it is not easy to calculate the KLD or the *F*-discrepancy apart from the state space. In what follows, we advocate a new metric that is insensitive to the dimension of the state space and is much easier to calculate for any particle set.

3.3 Sampling variance

The real number of times that each particle is resampled ($N_t^{(m)}$) must be an integer and is therefore often different from the expectation ($Nw_t^{(m)}$) even given that the resampling is unbiased such that $E(N_t^{(m)}) = Nw_t^{(m)}$. We define the sampling variance (SV) as the mean of the quadratic discrepancies between the numbers of times that the particles are resampled and the expectations:

$$\mathrm{SV} = \frac{1}{M}\sum_{m=1}^{M}(N_t^{(m)} - Nw_t^{(m)})^2. \tag{10}$$

The SV given in Eq. (10) is essentially a cost function that provides an efficient metric to measure the discrepancy between the discrete distributions of weighted particles before and after resampling. Based on the high-order moment of the discrepancy, it has a finer ability than the first statistical moment (e.g., the mean) to describe the discrepancy and should, therefore, be emphasized. The smaller the SV is, the better the ID quality of the resampling method is. If and only if the two distributions are exactly the same, the SV as well as the KLD or the *F*-discrepancy is zero. From this aspect, the SV measure is consistent with the KLD and the K-S statistic. They all measure the discrepancy of two distributions, differing in the statistics of interest. The next section gives the minimum condition for SV, resulting in a novel resampling algorithm.

4 Minimum-sampling-variance resampling

We now concern ourselves with optimizing the resampling algorithm under certain reasonable constraints. First, to minimize the variance of the weights (which to the largest extent reduces the degeneracy), the weight of the resampled particles shall be set equal, as commonly done; i.e.,

$$\tilde{w}_t^{(n)} = \frac{1}{N}. \tag{11}$$

We call this the optimal-weight condition, which fully removes the degeneracy. It is satisfied in most conventional resampling methods but possibly not in others, especially the compound sampling method (Li *et al.*, 2015). Under the optimal-weight condition, resampling is as simple as determining the number of times that each particle should be sampled, which reduces to an integer programming problem (Lenstra, 1983).

While satisfying Eq. (11), Eq. (10) can be minimized using a sampling scheme as given in Algorithm 1, where Floor(·) gives the largest integer not exceeding the content and TopRank$_s$[S] returns the largest s elements in set S. As is shown, the new resampling method is a deterministic sampling scheme, called minimum-sampling-variance (MSV) resampling, which consists of two main steps:

Step 1: Each particle is first resampled $\mathrm{Floor}(Nw_t^{(m)})$ times, leaving a weight residual $\hat{w}_t^{(m)}{=}w_t^{(m)} - \mathrm{Floor}(Nw_t^{(m)})/N$; this step will yield, in total, L particles, where $L{=}\sum_{m=1}^{M}\mathrm{Floor}(Nw_t^{(m)})$.

Step 2: The particle with relatively large weight residual, top $N-L$, will be further sampled one more time each.

Algorithm 1 MSV resampling

```
[$\{\tilde{x}_t^{(n)}\}_{n=1}^{N}$] = ResampleMSV[$\{x_t^{(m)}, w_t^{(m)}\}_{m=1}^{M}, N$];
$n$=0; $L$=0;
For $m$=1:$M$
    $N_t^{(m)} = \mathrm{Floor}(Nw_t^{(m)})$;
    $\hat{w}_t^{(m)} = w_t^{(m)} - N_t^{(m)}/N$;
    $L = L + N_t^{(m)}$;
End
For $m$=1:$M$
    If $\hat{w}_t^{(m)} \in \mathrm{TopRank}_{N-L}[\{\hat{w}_t^{(m)}\}_{m=1}^{M}]$
        $N_t^{(m)} = N_t^{(m)} + 1$;
    End
    For $h = 1 : N_t^{(m)}$
        $n$=$n$+1;
        $\tilde{x}_t^{(n)} = x_t^{(m)}$;
    End
End
```

The MSV resampling is guaranteed to achieve the minimum SV for an arbitrary sample set while satisfying the optimal-weight condition and achieving exactly the specified number of particles. This, however, comes at the price of relaxing the unbiasedness condition to be only asymptotically satisfied. We have Theorem 1 on the necessary condition to obtain MSV for resampling and Theorems 2–4 on the properties of the proposed MSV resampling approach; the proofs of these theorems are provided in Appendix B.

Theorem 1 The minimum condition of Eq. (10) requires $|N_t^{(m)} - Nw_t^{(m)}| < 1$ for any 1≤*m*≤*M*.

Theorem 2 The MSV resampling approach satisfies $|N_t^{(m)} - Nw_t^{(m)}| < 1$ for any 1≤*m*≤*M*.

Theorem 3 The MSV resampling approach minimizes SV.

Theorem 4 The MSV resampling approach is asymptotically unbiased.

The optimal ID quality obtained by MSV resampling indicates an ability to maximally preserve the posterior distribution or, to say, an ability to maximally reduce information loss in the process of resampling, which is highly preferable in theory.

Similarly, the so-called optimal sampling (OS) (Fearnhead and Clifford, 2003) and the stratified OS (Fearnhead and Liu, 2007) minimize the following discrepancy of the weight of particles before and after resampling:

$$\sum_{m=1}^{M} (\tilde{w}_t^{(m)} - w_t^{(m)})^2, \tag{12}$$

where $\tilde{w}_t^{(m)}$ is the new weight of particle $x_t^{(m)}$ if resampled and is equal to zero if not resampled.

The OS resampling, the detail of which can be found in the tutorial, attributes sample impoverishment to the replication of large-weighted particles and therefore, instead of replication, it reverses them. This is not so true since it is discarding (small-weighted) particles but not replication of particles that causes the problem. The particle weights after OS resampling can still have a high variance and suffer from degeneracy. In addition, the number of particles will surely decrease (i.e., *N*<*M*), which makes it suitable only for PFs with an increasing number of particles (e.g., new particles are added at some other steps). In contrast, the proposed MSV resampling avoids these problems while achieving the optimal SV.

It is also interesting to note that maintaining a stable/controllable (whether constant or adaptive) number of particles is critical for the PF in practice. For this, the largest discrepancy given in Eq. (12) was suggested to be bounded so that the smallest *N* can be determined (Fearnhead and Liu, 2007). This, however, is underestimated in the distributed PF developed by Sutharsan *et al.* (2012), where the number of particles based on a simple ‘rounding’ operation varies but is ignored throughout that study; please refer to Remark 1 of Li *et al.* (2013a), wherein a relevant resampling algorithm of a time-varying number of particles was proposed based on the ‘rounding’ operation. These indicate that resampling needs to be carefully treated, although it may be apparently simple in algorithm design.

5 Comparable study

A comprehensive quantitative comparison of the resampling methods is worth considering. Existing works such as those by Bashi *et al.* (2003), Hol *et al.* (2006), Murray (2012), Li *et al.* (2013a), and Sileshi *et al.* (2013) have compared no more than five resampling methods. In this section, we will compare numerous representative resampling methods based on a classic univariate state space model with a strong nonlinear state process equation and an observation equation given respectively by

$$x_t = \frac{x_{t-1}}{2} + \frac{25x_{t-1}}{1+x_{t-1}^2} + 8\cos(1.2(t-1)) + u_t, \tag{13}$$

$$y_t = \frac{x_t^2}{20} + v_t, \tag{14}$$

where x_t and y_t are the state and observation at time *t*, respectively, and Gaussian noises $u_t \sim \mathcal{N}(0, 10)$, $v_t \sim \mathcal{N}(0, 1)$.

The resampling methods used for simulation include single-sampling (including multinomial (Gordon *et al.*, 1993), stratified, systematic (Kitagawa, 1996), residual (Liu and Chen, 1998), RSR (Bolić *et al.*, 2003), branch-kill (Crisan and Lyons, 1999), and rounding-copy (Li *et al.*, 2013a)), reallocation (Liu *et al.*, 2001), deterministic resampling (Li *et al.*, 2012), simplified partial resampling (PR) (Hong *et al.*, 2010), KLD resampling (Li *et al.*, 2013b), Metropolis resampling (Murray, 2012), local selection resampling (Míguez *et al.*, 2004), and MSV resampling given in Algorithm 1. For the details of these algorithms, the readers are referred to the pseudocodes given in the tutorial (the MATLAB codes for these resampling algorithms can be found at https://sites.google.com/site/tianchengli85/matlab-codes/resampling-methods).

The simulation adopts different resampling methods but the same observation data at all time-steps and the same starting number of particles N_0 for each filter. All resampling algorithms maintain a constant number of particles that is equal to N_0, except KLD resampling, which automatically adapts its number of particles. The SIS, which does not use resampling, is also simulated. However, we omit here the selective resampling strategy that implements resampling only at selected steps. The filter estimate given by the weighted mean of all particles is extracted after resampling in each filter in order to directly reflect the impact of resampling.

Some resampling methods are parametric and their parameters are set as follows:

1. The length of grid (one-dimensional) used in the fixed-size grid-based deterministic resampling is R=1.

2. The length of grid (one-dimensional) used in KLD resampling is $\sqrt{R}/2=0.5$ and the number of particles in KLD resampling is determined online by

$$N=\frac{g-1}{2\varepsilon}\left(1-\frac{2}{9(g-1)}+\sqrt{\frac{2}{9(g-1)}}z_{1-\delta}\right)^3, \quad (15)$$

where g is the number of support grids that contain at least one particle, $z_{1-\delta}$ is the upper quartile of the standard normal distribution, ε=0.10, δ=0.01. In our case, N is further hard-limited to be no larger than $2N_0$.

3. For threshold-based simplified partial resampling, the threshold T=1/(5N) for Code 8 in the tutorial is used.

4. For Metropolis resampling, the threshold B=N/10 for Code 9 in the tutorial is used.

Results of these filters using different resampling methods are plotted in subsequent figures. In the figures, 'others' indicates all the resampling methods that have not been particularly specified in the legend. The true state and the filter estimates (when N_0 is assigned 100) are given in Fig. 1. The number of particles and the SV obtained using different resampling methods against time are given in Figs. 2 and 3, respectively. Here, we give only the SV but not the KLD or K-S statistic as it is much easier to compute. For 100 trials, the average root mean square error (RMSE) results are given in Fig. 4.

5.1 RMSE and SV

Regarding RMSE and SV, the key findings can be summarized in the following five points:

1. Resampling is critical for this filtering model as demonstrated by the obviously low accuracy of the SIS filter. In the following, we will not discuss the SIS filter further.

2. All unbiased resampling methods (including single sampling, reallocation, deterministic, and KLD resampling) and the asymptotically unbiased MSV resampling method, yield equivalent estimation accuracy in terms of RMSE (Fig. 4), especially when the number of particles is larger than 100. Even at each single run, their resampling results are almost the same (Fig. 1). This agrees with the qualitative study given in the tutorial, confirming that there is little difference between unbiased algorithms. The resampling methods of good ID quality, such as deterministic resampling and MSV resampling, do not show a significant advantage, except obtaining slightly better RMSEs when the number of particles is relatively small (e.g., <50). This indicates that the ID quality is more important for resampling when a small number of particles are used.

3. KLD resampling obtains better results when N_0<50 but worse results when N_0>80, as compared with unbiased (and asymptotically unbiased) methods. This is because the number of particles is adjusted online in KLD resampling, which is on average larger than N_0 when N_0<80 and smaller than N_0 when N_0>80. It internally performs random sampling, the same as multinomial resampling, but with an automatically determined number of particles and its superiority/inferiority is attributed only to the different numbers of particles used.

4. Local selection and Metropolis resampling produce obviously worse results than the others do. This is because they are significantly biased, which can deviate the estimate in this model. Their easy-to-parallelize advantage is not shown here and their bias may not lead to such bad results in other models; see,

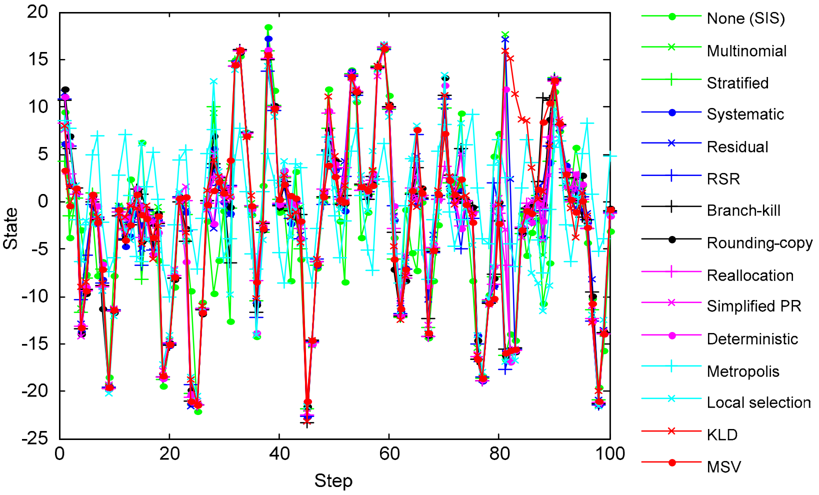

Fig. 1 True state and estimates against time when the starting number of particles is 100 (references to color refer to the online version of this figure)

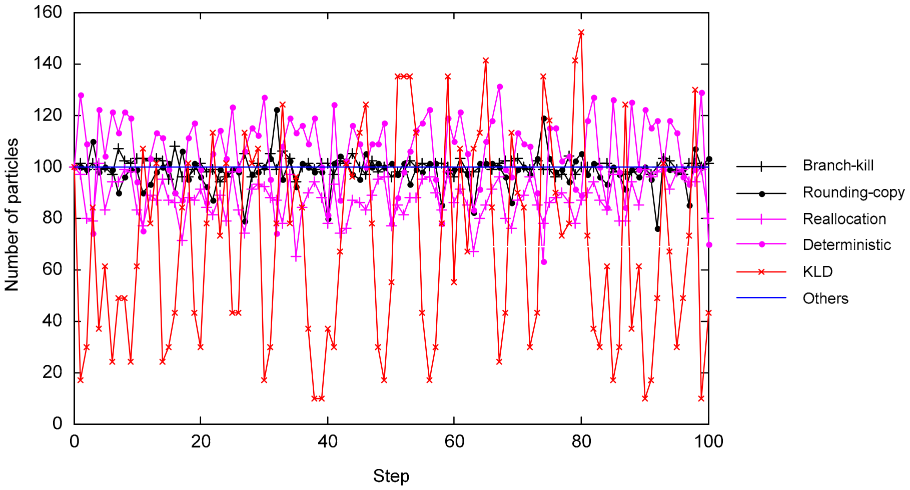

Fig. 2 Fluctuation of the number of particles against time (references to color refer to the online version of this figure)

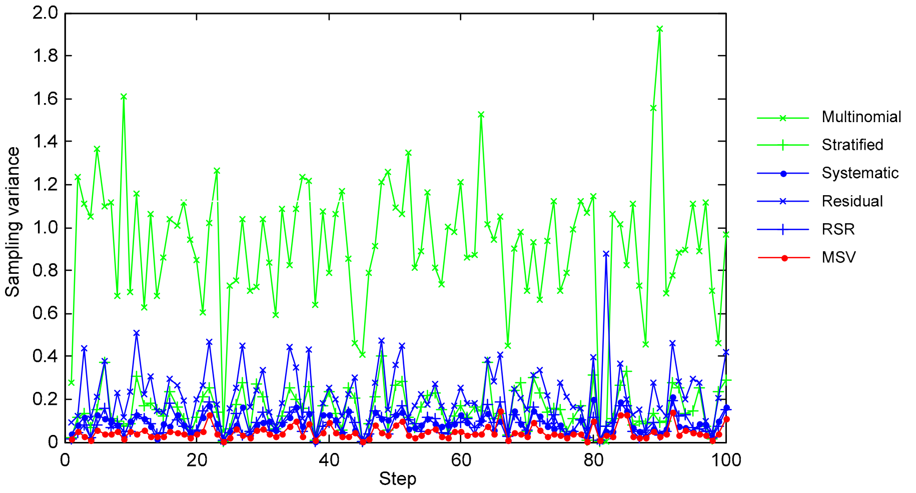

Fig. 3 Sampling variance of different resampling methods that satisfy the optimal weighting condition

e.g., Simonetto and Keviczky (2009). In addition, it is proven that parameter B is critical for Metropolis resampling, which is a trade-off between computing speed and reliability; $B=N/10$ is by no means the best choice for this model.

5. The proposed MSV resampling obtains the smallest SV (Fig. 3) among all methods that produce a constant number of equally weighted particles. This agrees with the theoretical justification.

5.2 Sample size

The fluctuation of the number of particles (namely, sample size) against time is plotted in Fig. 2, which shows that the number of particles varies in the range of 90–110 in both branch-kill and rounding-copy methods, in the range of 80–100 in the reallocation method, in the range of 80–120 in deterministic resampling, and in a larger range of 20–120 in KLD resampling. It remains constant in the other resampling methods for which no mechanism is designed to change the number of particles.

For a simulation period of 100 steps, the mean number of particles obtained by different resampling methods against the starting number of particles from 20 to 500 in 100 trials is given in Fig. 5. The following inferences can be ascertained:

1. The branch-kill, rounding-copy, deterministic resampling, and reallocation, in descending order, obtain the closest number of particles to N_0. Comparably, branch-kill performs the most stably. On average, rounding-copy has obtained a slightly smaller number of particles than the reference even though its mean is equal to the reference in theory (Li *et al.*, 2013a). We conjecture this is because the MATLAB software that uses limited-precision storage truncates the float number. The change of the number of particles in deterministic resampling is primarily due to its merging operation.

2. KLD resampling adjusts the number of particles online according to the system situation. In this highly nonlinear model, the state changes very sharply, so is the obtained number of particles. The advantage of adjusting the sample size online according to the system requirement can be very useful in real-time applications for complicated problem models, but it is not shown here.

5.3 Processing time

For the trial period of 100 steps, the mean processing time of different resampling methods against the starting number of particles N_0 from 20 to 500 in 100 trials is given in Fig. 6. It is worth noting that the computing speed depends on the hardware platform and the programming technology, for which all the resampling methods have been speeded up as equivalently as possible. Given this prerequisite, the results given in Fig. 6 indicate the following key findings:

1. With the same starting number of particles N_0, different resampling algorithms have significantly different computing speeds and they do not rank in the same order always. For example, when N_0 is very small, Metropolis (whose computing speed depends on parameter B) and multinomial resampling run very fast. However, by increasing N_0, the computing time required by them increases more significantly than for others. In addition, whether RSR (or reallocation) is faster or slower than the rounding-copy (or stratified resampling) method depends on the number of particles. This shall be carefully considered when choosing a resampling method.

2. Deterministic resampling and KLD resampling (when $N_0<300$) are the slowest since they need to create grids in the state space, which is computing-intensive. Next to them, simplified PR, local selection, and multinomial resampling (except very small sample size) are also slow. Roughly, the fewer the random number and the number of iterations used, the faster the speed of unbiased resampling.

3. Rounding-copy and RSR compute the fastest, followed by the branch-kill, systematic/stratified, and MSV resampling approximately in order, which use fewer/nil random numbers. This indicates that deterministic sampling is more computationally efficient than random sampling, while the former is also more efficient in gaining good ID property.

4. Comparably, KLD resampling maintains a relatively stable number of particles (Fig. 5) and therefore, its computational requirement will not increase with the number of particles (when $N_0>60$, the computing time of KLD resampling is fairly stable). In particular, when $N_0>350$, KLD resampling can be faster than multinomial resampling. The advantage of adjusting the sample size online according to the system requirement can be very useful in real-time applications in which likelihood calculation is computing-intensive, but it is not shown here.

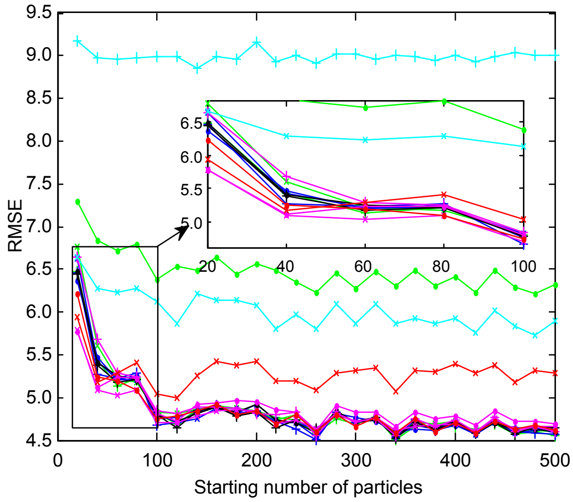

Fig. 4 Average RMSEs of particle filters against different starting numbers of particles

The legend is the same as that of Fig. 1. References to color refer to the online version of this figure

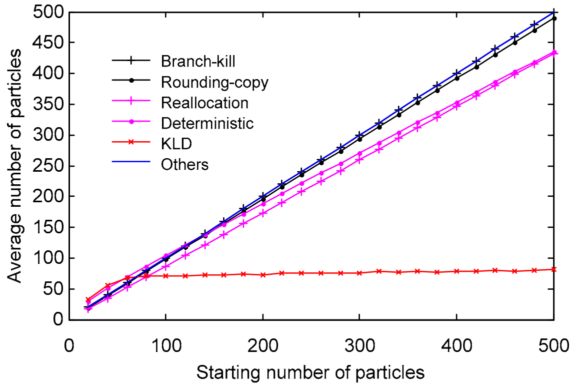

Fig. 5 Average numbers of particles of 100 steps of different particle filters (after resampling) against different starting numbers of particles

Overall, the simulation results demonstrate: First, the resampling methods can rarely produce very different results if they satisfy the unbiasedness (even asymptotically) condition, preserving a constant number of particles and equally weight resampled particles. Second, if these restrictions are removed, special benefits may be obtained, e.g., adaptively adjust the number of particles according to the system requirement, disregard unbiasedness in order to preserve particle diversity and thus to alleviate impoverishment or to enable parallel processing. However, these new advantages come at the price of more computational requirements (because of sophisticated algorithm design) and are generally model specific. Third, different resampling methods may have significantly different computational speeds, depending on the number of particles and the problem model. In general, deterministic and single-sampling algorithms are computationally faster than the random and compound sampling methods and are more suitable for parallel implementation.

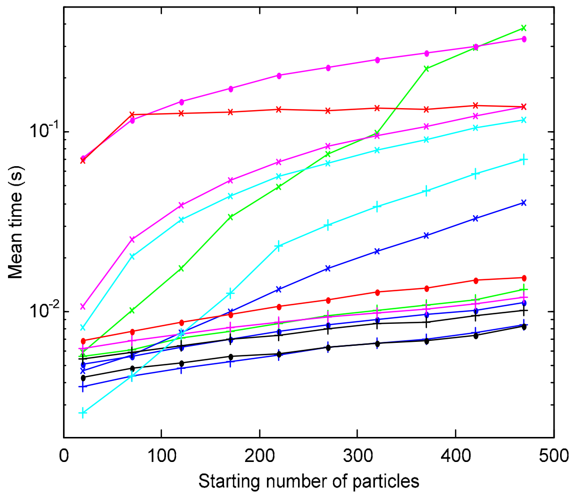

Fig. 6 Average processing time of 100 runs of different resampling methods against different starting numbers of particles

The legend is the same as that of Fig. 1. References to color refer to the online version of this figure

6 Conclusions

Resampling is an essential procedure for particle filtering, which is of both theoretical and practical importance. As a sequel to the tutorial (Li *et al.*, 2015), this paper has contributed in three more aspects. First, ID has been established as a fundamental principle for resampling, which can be measured by the KL divergence, K-S statistic, and the sampling variance. This affords a useful measure and perspective to compare and assess existing resampling methods or to design new methods. Second, following the ID principle, a simple albeit efficient resampling method, called MSV resampling, is proposed for general use that obtains the optimal ID attribute in terms of SV. Third, a comprehensive comparable study of more than a dozen representative resampling methods, including the proposed MSV resampling based on a classical state space model, is given. The results show that, most unbiased resampling methods do not exhibit much difference in terms of estimation accuracy (despite significantly biased resampling methods performing very badly), but they show significant differences in terms of SV and computing time and may provide special benefits in specific problems.

Acknowledgements

The authors acknowledge the insightful discussion with Prof. Petar DJURIĆ and Dr. Miodrag BOLIĆ on this work.

References

Adiprawita, W., Ahmad, A.S., Sembiring, J., *et al.*, 2011. New resampling algorithm for particle filter localization for mobile robot with 3 ultrasonic sonar sensor. Proc. Int. Conf. on Electrical Engineering and Informatics, p.1-6. [doi:10.1109/ICEEI.2011.6021733]

Arulampalam, M.S., Maskell, S., Gordon, N., *et al.*, 2002. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.*, **50**(2):174-188. [doi:10.1109/78.978374]

Bashi, A.S., Jilkov, V.P., Li, X.R., *et al.*, 2003. Distributed implementations of particle filters. Proc. 6th Int. Conf. on Information Fusion, p.1164-1171. [doi:10.1109/ICIF.2003.177369]

Beskos, A., Crisan, D., Jasra, A., 2014. On the stability of sequential Monte Carlo methods in high dimensions. *Ann. Appl. Probab.*, **24**(4):1396-1445. [doi:10.1214/13-AAP951]

Bolić, M., Djurić, P.M., Hong, S., 2003. New resampling algorithms for particle filters. Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, p.589-592. [doi:10.1109/ICASSP.2003.1202435]

Cappé, O., Godsill, S.J., Moulines, E., 2007. An overview of existing methods and recent advances in sequential Monte Carlo. *Proc. IEEE*, **95**(5):899-924. [doi:10.1109/JPROC.2007.893250]

Chen, Y., Xie, J., Liu, J.S., 2005. Stopping-time resampling for sequential Monte Carlo methods. *J. R. Stat. Soc. B*, **67**(2):199-217. [doi:10.1111/j.1467-9868.2005.00497.x]

Choe, G.M., Wang, T., Liu, F., *et al.*, 2014. An advanced integrated framework for moving object tracking. *J. Zhejiang Univ.-Sci. C (Comput. & Electron.)*, **15**(10): 861-877. [doi:10.1631/jzus.C1400006]

Choe, G.M., Wang, T., Liu, F., *et al.*, 2015. Particle filter with spline resampling and global transition model. *IET Comput. Vis.*, **9**(2):184-197. [doi:10.1049/iet-cvi.2014.0106]

Crisan, D., Doucet, A., 2002. A survey of convergence results on particle filtering methods for practitioners. *IEEE Trans. Signal Process.*, **50**(3):736-746. [doi:10.1109/78.984773]

Crisan, D., Lyons, T., 1999. A particle approximation of the solution of the Kushner-Stratonovitch equation. *Probab. Theory Related Fields*, **115**(4):549-578. [doi:10.1007/s004400050249]

Crisan, D., Del Moral, P., Lyons, T., 1998. Discrete Filtering Using Branching and Interacting Particle Systems. Markov Process. *Related Fields*, **5**(3):293-318.

Das, S.K., Mazumdar, C., 2013. Priori-sensitive resampling particle filter for dynamic state estimation of UUVs. Proc. 8th Int. Workshop on Systems, Signal Processing and Their Applications, p.384-389. [doi:10.1109/WoSSPA.2013.6602396]

Del Moral, P., Hu, P., Wu, L., 2012. On the concentration properties of interacting particle processes. *Found. Trends Mach. Learn.*, **3**(3-4):225-389. [doi:10.1561/2200000026]

Djurić, P.M., Miguez, J., 2010. Assessment of nonlinear dynamic models by Kolmogorov-Smirnov statistics. *IEEE Trans. Signal Process.*, **58**(10):5069-5079. [doi:10.1109/TSP.2010.2053707]

Djurić, P.M., Kotecha, J.H., Zhang, J., *et al.*, 2003. Particle filtering. *IEEE Signal Process. Mag.*, **20**(5):19-38. [doi:10.1109/MSP.2003.1236770]

Douc, R., Cappé, O., 2005. Comparison of resampling schemes for particle filtering. Proc. 4th Int. Symp. on Image and Signal Processing and Analysis, p.64-69. [doi:10.1109/ISPA.2005.195385]

Douc, R., Moulines, E., Olsson, J., 2014. Long-term stability of sequential Monte Carlo methods under verifiable conditions. *Ann. Appl. Probab.*, **24**(5):1767-1802. [doi:10.1214/13-AAP962]

Doucet, A., de Freitas, N., Gordon, N., 2001. Sequential Monte Carlo Methods in Practice. Springer, New York, USA. [doi:10.1007/978-1-4757-3437-9]

Efron, B., Rogosa, D., Tibshirani, R., 2015. Resampling methods of estimation. *In*: Wright, J.D. (Ed.), International Encyclopedia of the Social & Behavioral Sciences (2nd Ed.). Elsevier, Oxford, p.492-495. [doi:10.1016/B978-0-08-097086-8.42165-3]

Fearnhead, P., Clifford, P., 2003. On-line inference for hidden Markov models via particle filters. *J. R. Stat. Soc. Ser. B*, **65**(4):887-899. [doi:10.1111/1467-9868.00421]

Fearnhead, P., Liu, Z., 2007. On-line inference for multiple changepoint problems. *J. R. Stat. Soc. Ser. B*, **69**(4): 589-605. [doi:10.1111/j.1467-9868.2007.00601.x]

Fox, D., 2003. Adapting the sample size in particle filters through KLD-sampling. *Int. J. Robot. Res.*, **22**(12):985-1003. [doi:10.1177/0278364903022012001]

Godsill, S., Vermaak, J., Ng, W., *et al.*, 2007. Models and algorithms for tracking of maneuvering objects using variable rate particle filters. *Proc. IEEE*, **95**(5):925-952. [doi:10.1109/JPROC.2007.894708]

Gordon, N., Salmond, D., Smith, A.F.M., 1993. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc. F*, **140**(2):107-113. [doi:10.1049/ip-f-2.1993.0015]

Gustafsson, F., 2010. Particle filter theory and practice with positioning applications. *IEEE Aeros. Electron. Syst. Mag.*, **25**(7):53-82. [doi:10.1109/MAES.2010.5546308]

Hol, J.D., Schon, T.B., Gustafsson, F., 2006. On resampling algorithms for particle filters. Proc. IEEE Nonlinear Statistical Signal Processing Workshop, p.79-82. [doi:10.1109/NSSPW.2006.4378824]

Hong, S., Shi, Z., Chen, J., *et al.*, 2010. A low-power memory-efficient resampling architecture for particle filters. *Circ. Syst. Signal Process.*, **29**(1):155-167. [doi:10.1007/s00034-009-9117-4]

Hu, X.L., Schon, T.B., Ljung, L., 2011. A general convergence result for particle filtering. *IEEE Trans. Signal Process.*, **59**(7):3424-3429. [doi:10.1109/TSP.2011.2135349]

Kalman, R.E., 1960. A new approach to linear filtering and prediction problems. *J. Basic Eng.*, **82**(1):35-45. [doi:10.1115/1.3662552]

Kitagawa, G., 1996. Monte Carlo filter and smoother and non-Gaussian nonlinear state space models. *J. Comput. Graph. Stat.*, **5**(1):1-25. [doi:10.1080/10618600.1996.10474692]

Kong, A., Liu, J.S., Wong, W.H., 1994. Sequential imputations and Bayesian missing data problems. *J. Am. Stat. Assoc.*, **89**(425):278-288. [doi:10.1080/01621459.1994.10476469]

Kullback, S., Leibler, R.A., 1951. On information and sufficiency. *Ann. Math. Stat.*, **22**(1):79-86. [doi:10.1214/aoms/1177729694]

Kwak, N., Kim, G.W., Lee, B.H., 2008. A new compensation technique based on analysis of resampling process in FastSLAM. *Robotica*, **26**(2):205-217. [doi:10.1017/S0263574707003773]

Lang, H., Li, T., Villarrubia, G., *et al.*, 2015. An adaptive particle filter for indoor robot localization. Proc. 6th Int. Symp. on Ambient Intelligence, p.45-55. [doi:10.1007/978-3-319-19695-4_5]

Lenstra, H.W., 1983. Integer programming with a fixed number of variables. *Math. Oper. Res.*, **8**(4):538-548. [doi:10.1287/moor.8.4.538]

Li, T., Sun, S., 2010. Double-resampling based Monte Carlo localization for mobile robot. *Acta Autom. Sin.*, **36**(9):1279-1286. [doi:10.3724/SP.J.1004.2010.01279]

Li, T., Sattar, T.P., Sun, S., 2012. Deterministic resampling: unbiased sampling to avoid sample impoverishment in particle filters. *Signal Process.*, **92**(7):1637-1645. [doi:10.1016/j.sigpro.2011.12.019]

Li, T., Sattar, T.P., Tang, D., 2013a. A fast resampling scheme for particle filters. Proc. Constantinides Int. Workshop on Signal Processing, p.1-4. [doi:10.1049/ic.2013.0002]

Li, T., Sun, S., Sattar, T.P., 2013b. Adapting sample size in particle filters through KLD-resampling. *Electron. Lett.*, **46**(2):740-742. [doi:10.1049/el.2013.0233]

Li, T., Sun, S., Sattar, T.P., *et al.*, 2014. Fight sample degeneracy and impoverishment in particle filters: a review of intelligent approaches. *Expert Syst. Appl.*, **41**(8):3944-3954. [doi:10.1016/j.eswa.2013.12.031]

Li, T., Bolic, M., Djurić, P.M., 2015. Resampling methods for particle filtering: classification, implementation, and strategies. *IEEE Signal Process. Mag.*, **32**(3):70-86. [doi:10.1109/MSP.2014.2330626]

Li, T., Sun, S., Bolic, M., *et al.*, 2016. Algorithm design for parallel implementation of the SMC-PHD filter. *Signal Process.*, **119**:115-127. [doi:10.1016/j.sigpro.2015.07.013]

Liu, J.S., Chen, R., 1998. Sequential Monte Carlo methods for dynamic systems. *J. Am. Stat. Assoc.*, **93**(443):1032-1044. [doi:10.1080/01621459.1998.10473765]

Liu, J.S., Chen, R., Logvinenko, T., 2001. A theoretical framework for sequential importance sampling and resampling. *In*: Doucet, A., de Freitas, N., Gordon, N. (Eds.), Sequential Monte Carlo Methods in Practice. Springer, USA, p.225-246. [doi:10.1007/978-1-4757-3437-9_11]

Mbalawata, I.S., Särkkä, S., 2016. Moment conditions for convergence of particle filters with unbounded importance weights. *Signal Process.*, **118**:133-138. [doi:10.1016/j.sigpro.2015.06.018]

Míguez, J., Bugallo, M.F., Djurić, P.M., 2004. A new class of particle filters for random dynamical systems with unknown statistics. *EURASIP J. Adv. Signal Process.*, **15**:2278-2294. [doi:10.1155/S1110865704406039]

Morelande, M.R., Zhang, A.M., 2011. A mode preserving particle filter. Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, p.3984-3987. [doi:10.1109/ICASSP.2011.5947225]

Murray, L., 2012. GPU acceleration of the particle filter: the Metropolis resampler. arXiv:1202.6163v1.

Nielsen, F., 2010. A family of statistical symmetric divergences based on Jensen's inequality. arXiv:1009.4004.

Pérez, C.J., Martín, J., Rufo, M.J., *et al.*, 2005. Quasi-random sampling importance resampling. *Commun. Stat. Simul. Comput.*, **34**(1):97-112. [doi:10.1081/SAC-200047112]

Robert, C.P., Casella, G., 1999. Monte Carlo Statistical Methods. Springer, New York. [doi:10.1007/978-1-4757-4145-2]

Rubin, D.B., 1987. The calculation of posterior distribution by data augmentation: Comment: a noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when fractions of missing information are modest: the SIR algorithm. *J. Am. Stat. Assoc.*, **82**(398):543-546. [doi:10.2307/2289460]

Sileshi, B.G., Ferrer, C., Oliver, J., 2013. Particle filters and resampling techniques: importance in computational complexity analysis. Proc. Conf. on Design and Architectures for Signal and Image Processing, p.319-325.

Simonetto, A., Keviczky, T., 2009. Recent developments in distributed particle filtering: towards fast and accurate algorithms. Proc. 1st IFAC Workshop on Estimation and Control of Networked Systems, p.138-143. [doi:10.3182/20090924-3-IT-4005.00024]

Stano, P.M., Lendek, Z., Babuška, R., 2013. Saturated particle filter: almost sure convergence and improved resampling. *Automatica*, **49**(1):147-159. [doi:10.1016/j.

automatica.2012.10.006]

Sutharsan, S., Kirubarajan, T., Lang, T., *et al.*, 2012. An optimization-based parallel particle filter for multitarget tracking. *IEEE Trans. Aeros. Electron. Syst.*, **48**(2):1601-1618. [doi:10.1109/TAES.2012.6178081]

Topsoe, F., 2000. Some inequalities for information divergence and related measures of discrimination. *IEEE Trans. Inform. Theory*, **46**(4):1602-1609. [doi:10.1109/18.850703]

Wang, Y., Djurić, P.M., 2013. Sequential estimation of linear models in distributed settings. Proc. 21st European Signal Processing Conf., p.1-5.

Whiteley, N., 2013. Stability properties of some particle filters. *Ann. Appl. Probab.*, **23**(6):2500-2537. [doi:10.1214/12-AAP909]

Zhi, R., Li, T., Siyau, M.F., *et al.*, 2014. Applied technology in adapting the number of particles while maintaining the diversity in the particle filter. *Adv. Mater. Res.*, **951**:202-207. [doi:10.4028/www.scientific.net/AMR.951.202]

Appendix A: Assessment of different resampling methods in terms of the K-S statistic

A random selection process indicates that any particle can be missed randomly in the MR algorithm or be duplicated N times. Thus, the worst case (generating the largest F-discrepancy) is when the highest-weighted particle is abandoned (denoted with the subscript 'I'), i.e., $D_{\text{I,MR}} = \max(w_t^{(m)})$, or the smallest-weighted particle is duplicated N times (denoted with the subscript 'II'), i.e., $D_{\text{II,MR}} = 1 - \min(w_t^{(m)})$.

In the RR algorithm, random sampling is performed only to the weight residual. Therefore, we have $D_{\text{I,RR}} = \max(\hat{w}_t^{(m)})$, and $D_{\text{II,RR}} = N_2 / N - \min(\hat{w}_t^{(m)})$, where $N_2 = N - \sum_{m=1}^{M} Nw_t^{(m)}$. It is obvious that

$$\max(\hat{w}_t^{(m)}) < \frac{1}{N} \le \max(w_t^{(m)}), \tag{A1}$$

and

$$\min(\hat{w}_t^{(m)}) \le \min(w_t^{(m)}) < \frac{1}{N}. \tag{A2}$$

Then we obtain

$$D_{\text{I,MR}} > D_{\text{I,RR}}, \tag{A3}$$

$$D_{\text{II,MR}} > D_{\text{II,RR}}. \tag{A4}$$

This means that both 'I' and 'II' worst F-discrepancy of RR are smaller than that of MR.

Writing

$$D_{\text{MR}} = \max(D_{\text{I,MR}}, D_{\text{II,MR}}), \tag{A5}$$

$$D_{\text{RR}} = \max(D_{\text{I,RR}}, D_{\text{II,RR}}), \tag{A6}$$

we have

$$D_{\text{MR}} > D_{\text{RR}}. \tag{A7}$$

Since there is no random sampling in StR, SyR, RSR, and B-k algorithms, there is no so-called 'I' and 'II' worst cases as discussed above. However, the variation scope of the number of times that each particle will be resampled just indicates a level of the F-discrepancy; a larger varying scope indicates a larger discrepancy from the unbiased expectation. Therefore, according to Table III of the tutorial (Li *et al.*, 2015), we have Conclusion (9) as stated. We note these differences are slight.

Appendix B: Proofs of Theorems 1–4

Proof of Theorem 1 Without loss of generality, assuming the qth particle is resampled $N_t^{(q)}$ times satisfying $(N_t^{(q)} - Nw_t^{(q)}) \ge 1$, there must exist at least one particle $p: (N_t^{(p)} - Nw_t^{(p)}) < 1$ due to the overall condition $\sum_{m=1}^{M} (N_t^{(m)} - Nw_t^{(m)}) = 0$.

If we change the resampling results to be $N_{t,\text{new}}^{(p)} = N_t^{(p)} + 1$ and $N_{t,\text{new}}^{(q)} = N_t^{(q)} - 1$, while $N_{t,\text{new}}^{(i)} = N_t^{(i)}$, $i \ne p$, q, then we have the variance change as follows:

$$\begin{aligned}
&\text{Var}_{\text{new}} - \text{Var} \\
&= \frac{1}{M} \sum_{i=\{p,q\}} (N_{t,\text{new}}^{(i)} - Nw_t^{(i)})^2 - \frac{1}{M} \sum_{i=\{p,q\}} (N_t^{(i)} - Nw_t^{(i)})^2 \\
&= \frac{1}{M} \sum_{i=\{p,q\}} \left((N_{t,\text{new}}^{(i)})^2 - (N_t^{(i)})^2 + 2N_t^{(i)} Nw_t^{(i)} - 2N_{t,\text{new}}^{(i)} Nw_t^{(i)} \right) \\
&= \frac{1}{M} \Big((N_t^{(p)} + 1)^2 - (N_t^{(p)})^2 + 2N_t^{(p)} Nw_t^{(p)} \\
&\quad - 2(N_t^{(p)} + 1) Nw_t^{(p)} \Big) + \frac{1}{M} \Big((N_t^{(q)} - 1)^2 - (N_t^{(q)})^2 \\
&\quad + 2N_t^{(q)} Nw_t^{(q)} - 2(N_t^{(q)} - 1) Nw_t^{(q)} \Big) \\
&= \frac{2}{M} \left(1 + (N_t^{(p)} - Nw_t^{(p)}) - (N_t^{(q)} - Nw_t^{(q)}) \right) \\
&< \frac{2}{M} (1 + 0 - 1) = 0.
\end{aligned}$$

This indicates that in the resampling result of MSV, no particle is possible to be resampled $N_t^{(q)}$ times such that $(N_t^{(q)} - Nw_t^{(q)}) \ge 1$. A similar result holds in the case of $(N^{(q)} - Nw^{(q)}) \le -1$ and the proof is omitted here. Therefore, we conclude that the minimization of SV requires $|N_t^{(m)} - Nw_t^{(m)}| < 1$ for any 1≤*m*≤*M*.

Proof of Theorem 2 Given the sampling result $\{N_t^{(m)}\}_{m=1}^{M}$ obtained by the MSV resampling method as shown in Algorithm 1, it is obvious that, if

$$\hat{w}_t^{(m)} \in \text{TopRank}_{N-L}[\{\hat{w}_t^{(m)}\}_{m=1}^{M}],$$

then

$$N_t^{(m)}\text{=Floor}(Nw_t^{(m)})\text{+1};$$

otherwise, $N_t^{(m)}$=Floor($Nw_t^{(m)}$). Therefore, we have the results as stated.

Proof of Theorem 3 Theorems 1 and 2 indicate that the result of the MSV resampling approach and the minimum condition of SV are the same or they can become each other by adjusting $N_t^{(m)}$. If we can prove that any change in $N_t^{(m)}$ determined by the MSV resampling approach will cause an increase in SV, we will be certain that the MSV resampling approach achieves the minimum condition of SV.

Denote the numbers of times that these two particles are resampled according to the proposed approach as $N_t^{(p)}$ and $N_t^{(q)}$. Without loss of generality, we change the resampling results to be $N_{t,\text{new}}^{(p)}=N_t^{(p)}+l$ and $N_{t,\text{new}}^{(q)}=N_t^{(q)}-l$, where $1 \le l \le \min(N-L, N_t^{(q)})$, while $N_{t,\text{new}}^{(i)}=N_t^{(i)}$, *i≠p*, *q*; then, we have

$$\begin{aligned}
&\text{Var}_{\text{new}} - \text{Var} \\
&= \frac{1}{M}\sum_{i=\{p,q\}} (N_{t,\text{new}}^{(i)} - Nw_t^{(i)})^2 - \frac{1}{M}\sum_{i=\{p,q\}} (N_t^{(i)} - Nw_t^{(i)})^2 \\
&= \frac{1}{M}\sum_{i=\{p,q\}} \Big((N_{t,\text{new}}^{(i)})^2 - (N_t^{(i)})^2 + 2N_t^{(i)}Nw_t^{(i)} - 2N_{t,\text{new}}^{(i)}Nw_t^{(i)}\Big) \\
&= \frac{1}{M}\Big((N_t^{(p)}+l)^2 - (N_t^{(p)})^2 + 2N_t^{(p)}Nw_t^{(p)} \\
&\quad -2(N_t^{(p)}+l)Nw_t^{(p)}\Big) + \frac{1}{M}\Big((N_t^{(q)}-l)^2 - (N_t^{(q)})^2 \\
&\quad +2N_t^{(q)}Nw_t^{(q)} - 2(N_t^{(q)}-l)Nw_t^{(q)}\Big) \\
&= \frac{2l}{M}\Big(l + (N_t^{(p)} - Nw_t^{(p)}) - (N_t^{(q)} - Nw_t^{(q)})\Big).
\end{aligned} \tag{B1}$$

The proposed method satisfies the following bounds:

$$N_t^{(i)} - Nw_t^{(i)} = \begin{cases} -\hat{w}_t^{(i)}, & N_t^{(i)} \le Nw_t^{(i)}, \\ 1-\hat{w}_t^{(i)}, & N_t^{(i)} > Nw_t^{(i)}, \end{cases} \tag{B2}$$

and $0 < \hat{w}_t^{(p)}$, $\hat{w}_t^{(q)} < 1 \le l$. Specifically, if $N_t^{(p)} \le Nw_t^{(p)}$, $N_t^{(q)} > Nw_t^{(q)}$ (i.e., the latter is sampled once, while the former is not in the second part), then we have $\hat{w}_t^{(q)} > \hat{w}_t^{(p)}$. Therefore, Eq. (B1) will go to the following four cases:

1. If $N_t^{(p)} \le Nw_t^{(p)}$, $N_t^{(q)} \le Nw_t^{(q)}$, Eq. (B1) will reduce to

$$\text{Var}_{\text{new}} - \text{Var} = \frac{2l}{M}(l - \hat{w}_t^{(p)} + \hat{w}_t^{(q)}) > 0.$$

2. If $N_t^{(p)} \le Nw_t^{(p)}$, $N_t^{(q)} > Nw_t^{(q)}$, we have $\hat{w}_t^{(q)} > \hat{w}_t^{(p)}$ and Eq. (B1) will reduce to

$$\text{Var}_{\text{new}} - \text{Var} = \frac{2l}{M}(l - 1 + \hat{w}_t^{(q)} + \hat{w}_t^{(p)}) > 0.$$

3. If $N_t^{(p)} > Nw_t^{(p)}$, $N_t^{(q)} \le Nw_t^{(q)}$, Eq. (B1) will reduce to

$$\text{Var}_{\text{new}} - \text{Var} = \frac{2l}{M}(l - \hat{w}_t^{(p)} + \hat{w}_t^{(q)}) > 0.$$

4. If $N_t^{(p)} > Nw_t^{(p)}$, $N_t^{(q)} > Nw_t^{(q)}$, Eq. (B.1) will reduce to

$$\text{Var}_{\text{new}} - \text{Var} = \frac{2l}{M}(l - \hat{w}_t^{(p)} + \hat{w}_t^{(q)}) > 0.$$

It is shown that all these possible changes in the output of the MSV resampling method will lead to an increase of SV. This demonstrates that the proposed method outputs the minimum SV.

Proof of Theorem 4 For the proposed MSV approach, we have

$$\hat{w}_t^{(m)} = w_t^{(m)} - \frac{\text{Floor}(Nw_t^{(m)})}{N} < \frac{1}{N}. \tag{B3}$$

With the increase in the number of particles N, Eq. (B3) yields

$$\lim_{N\to\infty} \hat{w}_t^{(m)} = 0 \xrightarrow{\text{yields}} \lim_{N\to\infty} N_t^{(m)} = Nw_t^{(m)}. \tag{B4}$$

This indicates that the proposed approach is asymptotically unbiased. Furthermore, as shown in Theorem 2, the sampling bias is bounded by $|N_t^{(m)} - Nw_t^{(m)}| < 1$ for any 1≤m≤M.

Dr. Tian-cheng LI, first author of this invited research article, received his bachelor degree in mechanical and electrical engineering, with a minor in business administration, from Harbin Engineering University, China, in 2008, his PhD degree in electrical and electronic engineering from London South Bank University, UK, 2013, and his second PhD degree in mechanical and electrical engineering from Northwestern Polytechnical University, China, 2015. Since June 2014, he has been a postdoctoral researcher with the Bioinformatic, Intelligent Systems, and Educational Technology (BISITE) Group, University of Salamanca, Spain. His research interests lie in the general area of statistical signal processing, information fusion and distributed computing with an emphasis on particle filtering, and intelligent robot and multiple target tracking. He proposed two new estimation paradigms for multiple target detection and estimation (MODE): Clustering for Filtering (C4F) and Fitting for Smoothing (F4S).