Time dependent global optimization via Bayesian inference and Sequential Monte Carlo sampling

Piotr Kopka
1) National Centre for Nuclear Research, Świerk- Otwock, Poland
2) Institute of Computer Science of the Polish Academy of Sciences, Warsaw, Poland
Email: piotr.kopka@ncbj.gov.pl

Anna Wawrzynczak
1) National Centre for Nuclear Research, Świerk-Otwock, Poland
2) Institute of Computer Sciences, Siedlce Univesity, Poland
Email: a.wawrzynczak@ncbj.gov.pl

Mieczyslaw Borysiewicz
1) National Centre for Nuclear Research, Świerk-Otwock, Poland
Email: manhaz@ncbj.gov.pl

***Abstract*—In many areas of application it is important to estimate unknown model parameters in order to model precisely the underlying dynamics of a physical system. In recent years, Sequential Monte Carlo (SMC) methods have become a very popular tool for Bayesian parameter estimation. In this case, the problem of finding the best parameters configuration comes to the optimization issue which is to determine the best fit. In this paper, the application of this approach to the classical global optimization problem is described. We consider the situation when optimized functions are dynamical i.e. the global extremum is changing in time. For this purpose, we adapt two dimensional Ackley and four-dimensional Wood functions. Our aim is to find the most probable localization of the extremum in each time with the use of the Bayesian approach joined with the Markov Chain Monte Carlo (MCMC) and SMC algorithms. We propose a mechanism for dynamic tuning of the proposal distribution in SMC. The approach is based on the Metropolis-Hastings algorithm, combined with a resampling mechanism to achieve better results. We have examined different version of the proposed SMC and MCMC algorithms in terms of effectiveness to estimate the probabilistic distributions. The effect is demonstrated using two benchmark optimization problems. Computed results show that the proposed mechanisms can significantly improve optimization results compared to standard MCMC.**

I. Introduction

CONSIDER the general optimization problem (OP) designed with a time aspect i.e. the global extremum is changing its position with time (see e.g. Fig. 1). Suppose that various possibilities for a OP are defined by some parameters $\phi \in \Phi$, where Φ denotes the bounded space of parameters. As far as we have uncertainty connected with the best parameters configuration which provides optimum, we can express it in a form of a probability density function $P(\phi|D)$. $P(\phi)$ is the prior probability function that we can estimate based on the currently available information D [1]. Moreover, if new data D, related to the behavior of the optimization function, become available, it can be used for updating the prior distribution of searched parameters value $P(\phi)$ using Bayes theorem. This way we can obtain the posterior distribution $P(\phi|D)$ i.e. the distribution updated by the new information. In design optimization algorithms, the goal is to find the optimal values of the parameters set that minimizes (maximizes) the considered function. We consider that effectiveness of searching for the function extremum in subsequent time step can be increased by taking advantage from the information about the location of extremum in previous stages.

Previously, we have applied the methodology combining Bayesian inference with Markov Chain Monte Carlo (MCMC) methods to the problem of the contaminant source localization based only on the substance concentrations registered by distributed sensors network ([2] and [3]).

In this paper, we propose the application of the Sequential Monte Carlo (SMC) methods combined with the Bayesian inference to the global optimization problem. We present the possibility to connect MCMC and SMC to provide additional benefit in the process of event reconstruction. Proposed algorithms were tested on two benchmark functions where optimum was moving with time.

II. Optimization algorithm theoretical preliminaries

A. Bayesian inference

A good introduction to Bayesian theory can be found in [4] and [5]. Bayes' theorem, as applied to optimization problem:

$$P(\phi|D) = \frac{P(D|\phi)P(\phi)}{P(D)} \tag{1}$$

where ϕ represents possible configuration of optimization function parameters and D is the value of the optimized function at given point.

For our problem, Bayes' theorem describes the conditional probability $P(\phi|D)$ of optimum parameters (configurations of variables ϕ) given observed value of function under consideration (D). This conditional probability $P(\phi|D)$ is also known as the posterior distribution and is related to the probability of the D conforming to a given parameters configuration $P(D|\phi)$, and to the possible model configurations $P(\phi)$, before updating by new information D. The probability $P(D|\phi)$, for fixed D, is called the likelihood function, while $P(\phi)$ is the prior distribution. $P(D)$ is the marginal distribution of D and is called prior predictive distribution. $P(D)$ serves as a scaling factor and is this case is equal 1. So, in our case the

978-1-4673-4471-5/$25.00 © 2013, IEEE

Bayes theorem can be written as follows:

$$P(\phi|D) \propto P(D|\phi)P(\phi) \tag{2}$$

To estimate the unknown function's optimum parameters ϕ using (2), the posterior distribution $P(D|\phi)$ must be sampled. $P(D|\phi)$ quantifies the likelihood of a set of measurements D given the function's optimum parameters ϕ.

We use a sampling procedure with the Metropolis-Hastings algorithm to obtain the posterior distribution $P(\phi|D)$. This way we completely replace the Bayesian formulation with a stochastic sampling procedure to explore the optimized function parameters' space and to obtain a probability distribution for the optimum location.

B. The likelihood function

A measure indicating the quality of the current state of Markov chain is expressed in terms of a likelihood function. This function is proportional to considered global optimization function $H(.)$:

$$ln[P(D|\phi)] = ln[\lambda(\phi)] \propto H(\phi) \tag{3}$$

After calculating value of the likelihood function for the proposed state its acceptance is performed as follows:

$$\frac{ln(\lambda_{prop})}{ln(\lambda)} \geq U(0,1) \tag{4}$$

where λ_{prop} is the likelihood value of the proposal state, λ is the previous likelihood value, and $U(0,1)$ is a random number generated from a uniform distribution in the interval $(0,1)$.

It is important to note that condition (4) is more likely to be satisfied if the likelihood of the proposal is only slightly lower than the previous likelihood value. It gives a chance to choose even a little "worse" state, because the probability of acceptance depends directly on the quality of proposed state.

C. Posterior distribution

The posterior probability distribution (2) is computed directly from the resulting samples defined by the algorithm described below and is estimated with

$$P(\phi|D) \equiv \hat{\pi}^N(\phi) = \frac{1}{N}\sum_{i=1}^{N}\delta(\phi_i - \phi). \tag{5}$$

$P(\phi|D)$ represents the probability of a particular parameters configuration ϕ. Equation (5) is a sum over the entire samples set of length N of all the sampled values ϕ_i. Thus $\delta(\phi_i - \phi) = 1$ when $\phi_i = \phi$ and 0 otherwise. Consequently, if a Markov chain spends several iterations at the same location value of $P(\phi|D)$ increases through the summation (increasing the probability for those optimum parameters).

D. Sequential Monte Carlo

Sequential Monte Carlo (SMC) is designed to sample from dynamic posterior distributions. The SMC methods are easy to parallelize - the different Monte Carlo proposals can be generated and evaluated in parallel. A good introduction to SMC is present in [6], [7], [8] .

E. Sequential importance resampling

Sequential importance resampling (SIR) is a sequential version of importance sampling (IS) and combines IS with resampling procedure [9] . At the center of the SMC approach in our case is the generation of a weighted sample using IS method. IS uses a proposal distribution $q(.)$, that is close to target distribution $\pi(.)$ and from which it is easy to generate samples. The basic methodology is given below.

1) Generate a sample of size N from the proposal distribution $q(\phi)$:
$$\phi_{(i)} \sim q(\phi), i = 1, ..., N \tag{6}$$
2) Compute the importance weights:
$$\breve{w}(\phi_{(i)}) \propto \frac{\pi(\phi_{(i)})}{q(\phi_{(i)})}, i = 1, ..., N \tag{7}$$
and define
$$w(\phi_{(i)}) = \frac{\breve{w}(\phi_{(i)})}{\sum_{j=1}^{N}\breve{w}(\phi_{(j)})} \tag{8}$$
3) The distribution $\pi(\cdot)$ is then approximated by
$$\breve{\pi}^N(\phi) \equiv \sum_{i=1}^{N} w(\phi_{(i)})\delta(\phi_i - \phi) \tag{9}$$
which places the probability mass $w(\phi_{(1)}), ..., w(\phi_{(N)})$ on the support points $\phi_{(1)}, ..., \phi_{(N)}$.

Hence, the weights would be proportional to the value of likelihood. In our case to calculate the weight we use of the following formula,which is related to the likelihood function (3):

$$\breve{w}(\phi_{(i)}) \propto \frac{1}{ln[\lambda(\phi_{(i)})]}, i = 1, ..., N \tag{10}$$

Resampling is used to avoid the situation when almost all (except only a few) of the importance weights are close to zero (problem of degeneracy of the algorithm). Basic idea of resampling methods is to eliminate samples which have small normalized importance weights and to concentrate upon samples with large weights. So,:

1) for $i = 1, ..., N$ are chosen samples with indexes $k(i)$ distributed according to the discrete distribution with N elements satisfying
$$P(k(i) = l) = w(\phi_{(i)}) \tag{11}$$
for $l = 1, ..., N$,
2) then for $i = 1, ..., N$ for samples $M_{k(i)}$ are assigned the weights
$$w(\phi_{k(i)}) = \frac{1}{N}. \tag{12}$$

A sufficient number of draws is called Effective Sample Size (ESS) and is equal:

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N} w(\phi_{(i)})^2}. \tag{13}$$

where $w(\phi_{(i)})$ are normalized weights. If all weights are equal $1/N$ then effective sample size is N. In the contrast to a situation where all weights $= 0$, except for one weight $= 1$, effective sample size is equal 1.

F. MCMC prior to SMC

The SMC algorithm needs some set of samples to be initialized. An ideal way to generate this initial sample is using MCMC data from first K iterations in all time steps. The resulting equally weighted MCMC set of samples can then be passed on to SMC for processing in the subsequent iteration.

First, the scanning algorithm starts from the randomly chosen values of parameters ϕ (i.e. first we start from the "flat" priori). This assumption reflects lack of knowledge about the function optimum parameters. For the actual state ϕ likelihood function λ is calculated. Then we apply random walk procedure "moving" our Markov chain to the new position. Precisely, we change each model ϕ parameter by the value draw from the Gaussian distribution with the zero mean and variance σ^2_ϕ each parameter. Standard deviations for sampling parameters are determined by the problem's domain size and refined with a trial and error procedure to ensure that the Markov chains had access to realistic ranges with minimal occurrences of stuck problem. Problem of stuck in chains can occur when the standard deviations chosen for the next iteration lead to a large number of rejected samples, causing that the chain remains in a given position for many iterations. For the proposal state the likelihood function λ_{prop} is again estimated. We compare this two values λ and λ_{prop} according to (4). If comparison is more favorable than the previous chain location, the proposal is accepted (Markov chain "moves" to the new location). If the comparison is "worse", new state is not immediately rejected. Random variable from binomial distribution is used to decide whether or not to accept the new state of chain. After K iteration we pass all the samples (from all m chains)to the sequential procedure. We compute importance weights by (10) and normalize them. Next we use roulette procedure to draw N samples from the set generated by Markov Chain.

This random component is important because it prevents the chain from becoming trapped in a local minimum. The pseudo code for one time step of the algorithm is given below.

One of the important aspects of stochastic procedure of calculating the posterior distribution is choosing burn-in phase. The burn-in factor represents the number of samples needed at the beginning for the Markov chain to actually reach the search state where it is sampling from the target distribution.

Statistical convergence (to the posterior distribution) is monitored by computing between-chain variance and within-chain variance [4]. If there are m Markov chains of length N, then we can compute between-chain variance B with

$$B = \frac{N}{m-1}\sum_{j=1}^{m}(\bar{\phi_j} - \bar{\phi})^2 \tag{14}$$

where $\bar{\phi_j}$ is the average value along each Markov and $\bar{\phi}$ is the average of the values from all Markov chains. The within-chain variance W is

$$W = \frac{1}{m}\sum_{i=1}^{m} s_i^2 \tag{15}$$

where

$$s_i^2 = \frac{1}{N-1}\sum_{i=1}^{N}(\phi_{ij} - \bar{\phi_i})^2 \tag{16}$$

The convergence parameter R is then computed as

$$R = \frac{var(\phi)}{W} \tag{17}$$

where $var(\phi)$ is estimate variance of ϕ and is computed as

$$var(\phi) = \frac{N-1}{N}W + \frac{1}{N}B. \tag{18}$$

In this paper, we consider the following variants of scanning algorithms:

1) **Classic MCMC**
In this algorithm, the parameter space scan in each time step t is independent form the previous ones. So, in this case we don't use information from past calculations. Classic MCMC don't use sequential mechanism.

2) **SMC via Maximal Weights**
As the first location of Markov chain ϕ_0^t it select the set of ϕ parameters for which weight in previous time step procedure was the highest. So, for $t > 1$:

$$\phi_0^t \sim \ arg\ (\phi \in \{\phi_0^{t-1}, ..., \phi_n^{t-1}\})\ max\{w(\phi_i^{t-1})\} \tag{19}$$

With this approach, we always start with the best values found so far.

3) **SMC via Rejuvenation and Extension**
In contrast to SMC via Maximal Weights this algorithm as the first location of Markov chain ϕ_0^t at the time $t > 1$ chooses the set of parameters ϕ selected randomly from previous realization of resampling procedure in $t-1$ with use of the uniform distribution:

$$\phi_0^t \sim U(\phi_0^{t-1}, \phi_1^{t-1}, ..., \phi_n^{t-1}) \tag{20}$$
$$a\ uniform\ distribution\ \{1, ..., n\}$$

Applying the new knowledge (new measurements) the current chain is "extended" starting from selected position with use of the new information in the likelihood function calculation.

III. Algorithms results for selected optimization problem

A. Two-dimensional (2D) Ackley function

We have benchmark the proposed global optimization algorithms with use of the 2D version of Ackley function Fig. 2.

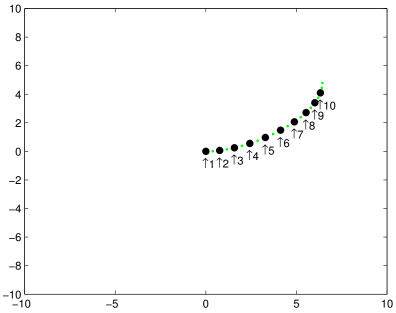

Fig. 1. Trajectory of the optimum of Ackley function

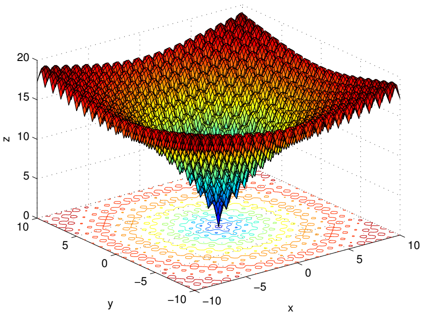

Fig. 2. Surf of two-dimensional Ackley benchmark function

$$H_1(x, y) = -20 \exp(-0.2\sqrt{0.5(x^2 + y^2)}) - \exp(0.5(cos(2\pi x) + \cos(2\pi y))) + 20 + e \quad (21)$$

The proposed optimization algorithms are designed to search for the optimum of the dynamical functions. To achieve the dynamical nature of the process described by the considered function we have ascribe the displacement of the optimum in 10 subsequent time steps. The assumed trajectory of the searched function optimum is presented in Fig. 1.

Based on the dynamical Ackley function we would like to compare the performance of two described in previous chapter SMC algorithms (i.e. SMC via Maximal Weights and SMC via Rejuvenation and Extension) in compare with a well-known stochastic simulation method i.e. classic MCMC. Since we are interested in runtime of all algorithms for optimization problems we use exactly the same parameters. The number of iteration for each algorithm is equal $K = 2000$. This number was chosen based on the numerical experiments as the number of iteration needed to reach convergence for each sampled dimension ($R \approx 1$) Fig. 3. The same way we tuned the rest of the algorithm parameters which adequately are equal: number of chains M=10; burn-in factor=500.

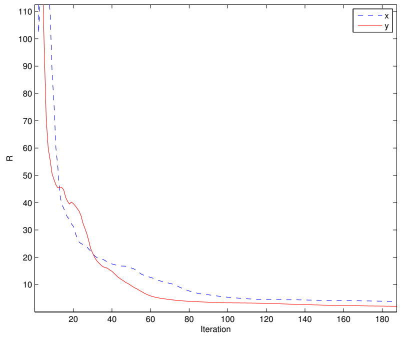

Fig. 3. R convergence parameter for x and y. The samples came from results of MCMC algorithm.

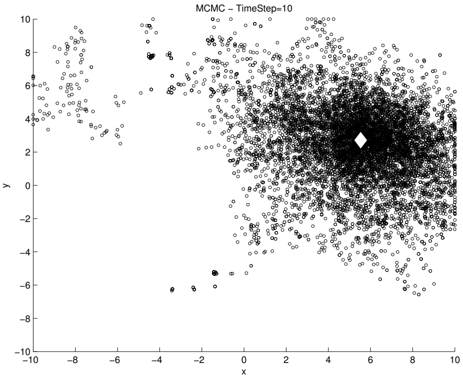

Fig. 4. The traces of three Markov chains in the x,y space. The global minimum is marked by diamond. The samples came from results of MCMC algorithm.

Fig. 5, and 6 presents the probability distributions of x and y optimum parameters in each time step for classic MCMC algorithm. Fig. 7 and 8 presents the same results for SMC via Maximal Weights and Figs. 10 and 11 for SMC via Rejuvenation and Extension, respectively. The target value of search optimum parameters are denoted by vertical lines. One can see that for the 2D Ackley dynamical function all algorithms successfully generate samples in the vicinity of the optimal solution. If we look carefully we can denote the difference of its probability values for the search parameters x and y. Moreover, the posteriori distributions of MCMC algorithm are much flatter than for SMC algorithms. For SMC via Maximal Weights and SMC via Rejuvenation and Extension the maximum value of the probability distribution x and y is close to 0.05 while for MCMC it is ≈ 0.027.

In both SMC algorithms transmission of the information in subsequent time steps about ”fleeing minimum” effect enlarge the concentration of the samples around the target optimum. Resampling mechanism can be seen in Fig. 4 and Fig. 9. Fig. 4 presents the traces of the Markov chains for classic MCMC and Fig. 9 for SMC via Rejuvenation and Extension in the last time step. One can see that MCMC algorithm consider samples spread out far from the the searched optimum values, at the same time the SMC method in subsequent time steps choose samples close to the target value, which results in the increase of its probability.

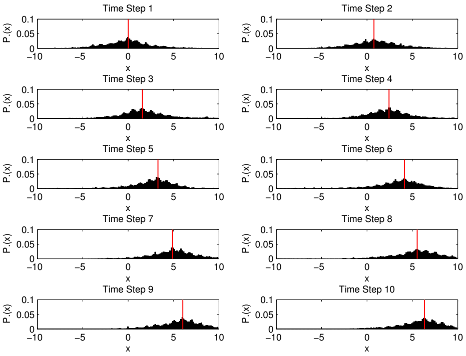

Fig. 5. Posterior distribution of x parameter in subsequent time steps for MCMC algorithm. Vertical line represents the target value of x.

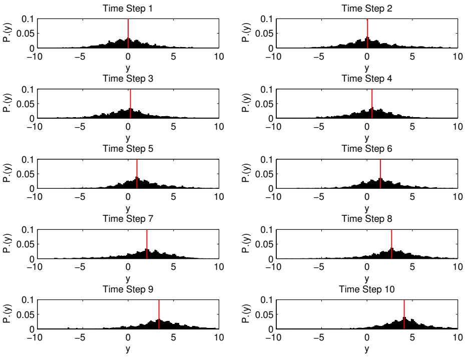

Fig. 6. Posterior distribution of y parameter in subsequent time steps for MCMC algorithm. Vertical line represents the target value of y.

B. Optimization problem - 4D Wood function

The previous example showed that the SMC algorithms give impute to the value of the probability of the searched optimum parameters (the probability is doubled). However, the classic MCMC also reached the target value of optimum. We would like to check if the proposed algorithms increase their efficiency in the case of the multidimensional space.

To test the effectiveness of SMC for optimization problem

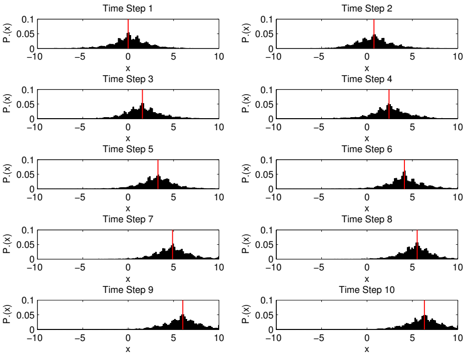

Fig. 7. Posterior distribution of x parameter in subsequent time steps for SMC via Maximal Weights. Vertical line represents the target value of x.

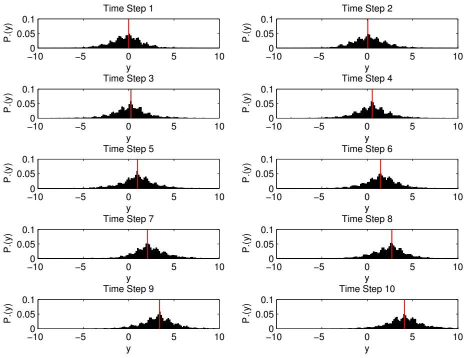

Fig. 8. Posterior distribution of y parameter in subsequent time steps for SMC via Maximal Weights. Vertical line represents the target value of y.

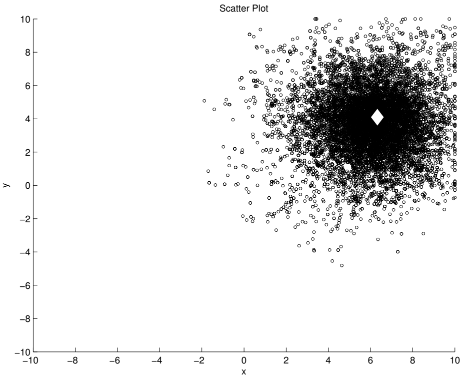

Fig. 9. Scatter plot of samples in the x,y space. The global minimum is marked by diamond. The samples came from results of SMC via Rejuvenation and Extension

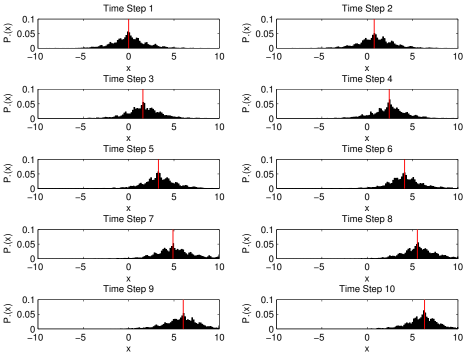

Fig. 10. Posterior distribution of x parameter in subsequent time steps for SMC via Rejuvenation and Extension. Vertical line represents the target value of x.

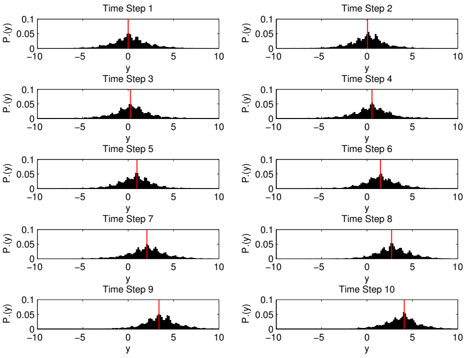

Fig. 11. Posterior distribution of y parameter in subsequent time steps for SMC via Rejuvenation and Extension. Vertical line represents the target value of y.

with higher dimensions, we consider the four-dimensional (4D) Wood function in our second benchmark test:

$$H_2(x_1, x_2, x_3, x_4) = 100({x_1}^2 - x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90({x_3}^2 - x_4)^2 + 10.1((x_2 - 1)^2 + (x_4 - 1)^2) + 19.8(x_2 - 1)(x_4 - 1) \tag{22}$$

In this test we also assume that the optimum initial value $x = (1, 1, 1, 1)$ moves in 6 subsequent time steps reaching at last $x = (3.84, 6.89, 3.84, 6.89)$. In this test for all considered algorithms we take: number of iteration $K = 20000$, number of chains $M = 10$; burn-in factor = 2000.

Figs. 12- 22 presents the marginal probability distributions for all four optimum parameters of the considered 4D Wood dynamical function. The target minimum location in each dimension is marked by the vertical red line. One can see that for the 4D Wood function efficiency of the classic MCMC is decreased. This method do not mark the target value of x_2 and x_4 parameters as the values with the highest probability (Figs. 13, 15). At the same time the results obtained from the two SMC algorithms are better than the MCMC algorithm. However, one can note than the SMC via Rejuvenation and Extension algorithm seems to be more efficient than SMC via Maximal Weights. The reason is that the SMC via Maximal Weights results for parameter x_2 (Fig. 17)are a bit worse than obtained from SMC via Rejuvenation and Extension (Fig. 21). Moreover, the SMC via Rejuvenation and Extension denotes the target values of the x_1 ,x_3 and x_4 parameter with higher probabilities than SMC via Maximal Weights.

It is worth to mention that SMC via Maximal Weights, SMC via Rejuvenation and Extension use the probability distributions obtained based on information from previous time steps to update the probability distributions with use of the new information. This causes a significant increase in convergence of the algorithm to the target location of the function's optimum in the subsequent time steps. This methodology makes these algorithms more effective for optimization of multidimensional dynamical functions than classic MCMC.

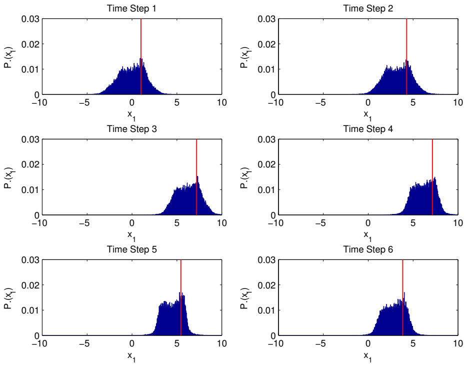

Fig. 12. Posterior distribution of x_1 parameter in subsequent time steps for MCMC. Vertical line represents the target value of x_1.

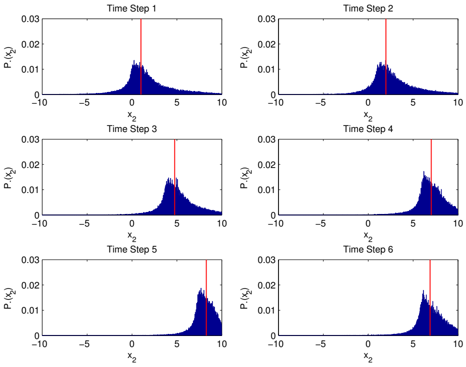

Fig. 13. Posterior distribution of x_2 parameter in subsequent time steps for MCMC. Vertical line represents the target value of x_2.

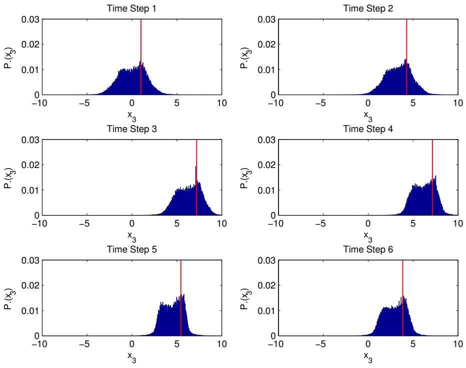

Fig. 14. Posterior distribution of x_3 parameter in subsequent time steps for MCMC. Vertical line represents the target value of x_3.

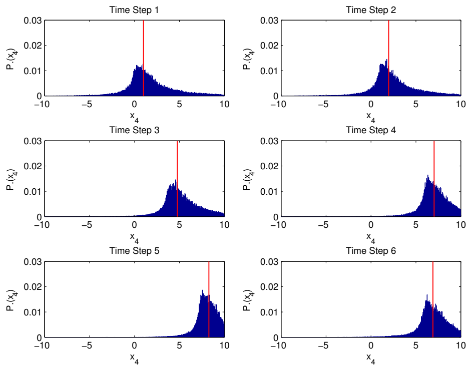

Fig. 15. Posterior distribution of x_4 parameter in subsequent time steps for MCMC. Vertical line represents the target value of x_4.

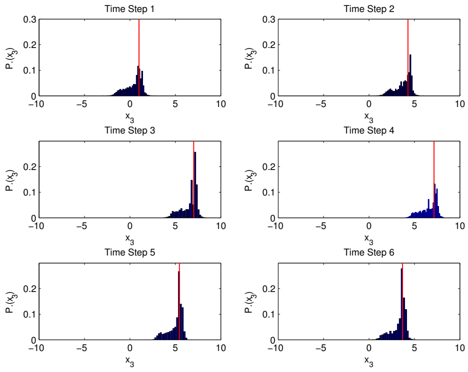

Fig. 18. Posterior distribution of x_3 parameter in subsequent time steps for SMC via Maximal Weights. Vertical line represents the target value of x_3.

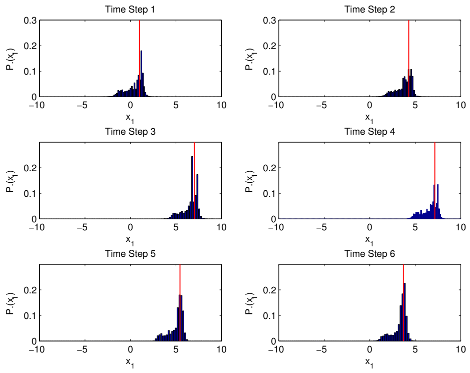

Fig. 16. Posterior distribution of x_1 parameter in subsequent time steps for SMC via Maximal Weights. Vertical line represents the target value of x_1.

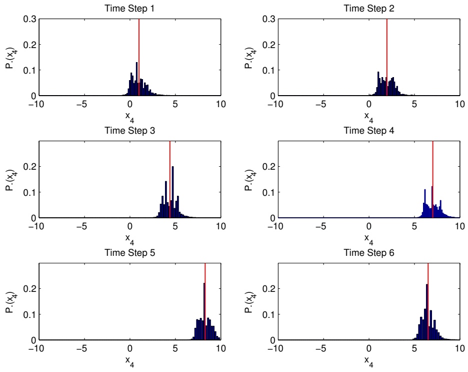

Fig. 19. Posterior distribution of x_4 parameter in subsequent time steps for SMC via Maximal Weights. Vertical line represents the target value of x_4.

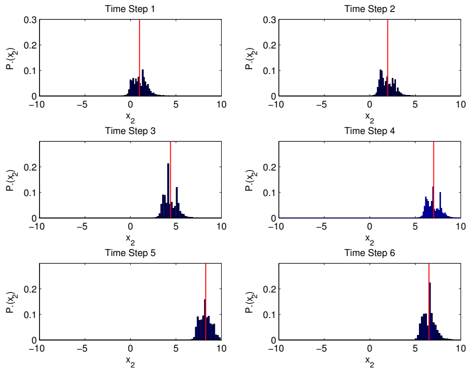

Fig. 17. Posterior distribution of x_2 parameter in subsequent time steps for SMC via Maximal Weights. Vertical line represents the target value of x_2.

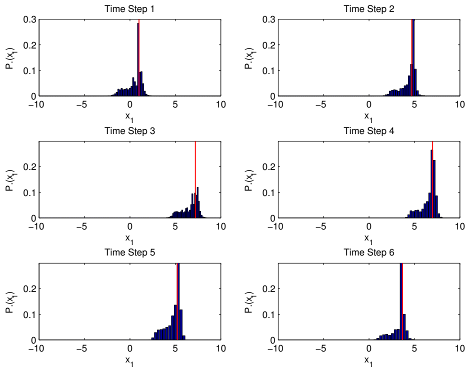

Fig. 20. Posterior distribution of x_1 parameter in subsequent time steps for SMC via Rejuvenation and Extension. Vertical line represents the target value of x_1.

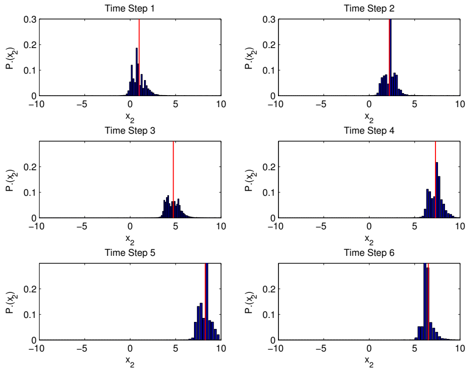

Fig. 21. Posterior distribution of x_2 parameter in subsequent time steps for SMC via Rejuvenation and Extension. Vertical line represents the target value of x_2.

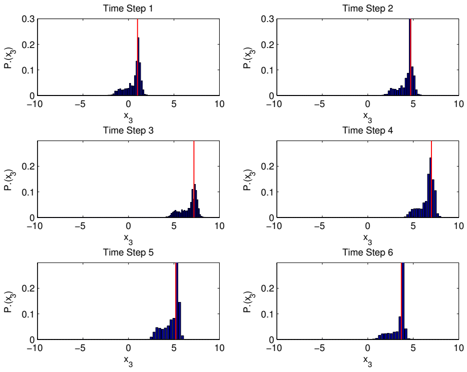

Fig. 22. Posterior distribution of x_3 parameter in subsequent time steps for SMC via Rejuvenation and Extension. Vertical line represents the target value of x_3.

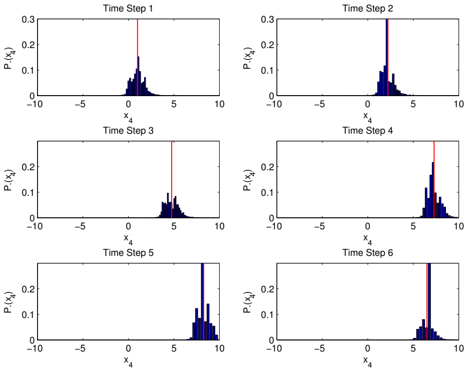

Fig. 23. Posterior distribution of x_4 parameter in subsequent time steps for SMC via Rejuvenation and Extension. Vertical line represents the target value of x_4.

IV. Conclusion

We have presented a methodology to solve the global optimization problem of dynamical functions with use of the Bayesian approach joined with SMC algorithms. The presented method combines Bayesian inference with SMC sampling and produces posterior probability distributions of the searches extremum's parameters. We have examined two version of the SMC algorithms i.e. SMC via Maximal Weights, SMC via Rejuvenation and Extension and compare its efficiency to estimate the probabilistic distributions of optimum parameters for 2D and 4D optimization functions. We compared the effectiveness of the proposed SMC algorithms with classic MCMC and have shown the advantage of the SMC algorithms that in different ways use the probability distributions of possible optimum parameters obtained basing on samples generated in previous time steps. We have shown that efficiency of proposed SMC algorithms increases with increasing the dimension of the optimized dynamical function. We conclude that proposed methodology joining the Bayesian inference with SMC algorithms is effective for optimization of multidimensional dynamical functions.

Acknowledgment

This work was supported by the Welcome Programme of the Foundation for Polish Science operated within the European Union Innovative Economy Operational Programme 2007-2013 and by the EU and MSHE grant nr POIG.02.03.00-00-013/09; Project VI.B.08 financed by the National Centre for Research and Development.

References

[1] Zuev, K.M., Beck, J.L., (2013): Global optimization using the asymptotically independent Markov sampling method. *Comput Struct*, http://dx.doi.org/10.1016/j.compstruc.2013.04.005

[2] Borysiewicz, M., Wawrzynczak A.,Kopka P., (2012): Stochastic algorithm for estimation of the model's unknown parameters via Bayesian inference. *Proceedings of the Federated Conference on Computer Science and Information Systems, 501–508, IEEE Press*,

[3] Borysiewicz, M., Wawrzynczak, A., Kopka, P., (2012): Bayesian-Based Methods for the Estimation of the Unknown Model's Parameters in the Case of the Localization of the Atmospheric Contamination Source, *Foundations of Computing and Decision Sciences*, 37, 4, 253–270

[4] Gelman, A., Carlin, J., Stern, H., Rubin, D., (2003): Bayesian Data Analysis. *Chapman & Hall/CRC*, 668 pp.

[5] Gilks, W., Richardson, S., Spiegelhalter ,D., (1996): Markov Chain Monte Carlo in Practice. *Chapman & Hall/CRC*, 486

[6] Doucet, A.,J. F. G.de Freitas, and Gordon, N. J., (2001): Sequential Monte Carlo methods in practice. *New York: Springer-Verlag*

[7] Pitt, M. K. Shephard, N. (2001): Sequential Monte Carlo methods in practice, chapter Auxiliary variable based particle filters. *New-York: Springer-Verlag*

[8] Liu, J. S. (2001): Monte Carlo Strategies in Scientific Computing. *New York: Springer*

[9] Gordon, N. J., Salmond, D. J., Smith, A. F. M., (1993): Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings F on Radar and Signal Processing* 140 (2): 107–113,