International Journal of Database Theory and Application
Vol.8, No.4 (2015), pp.117-122
http://dx.doi.org/10.14257/ijdta.2015.8.4.12

Dynamic Guaranteed Cost Compression for Time Series Big Data

Miao Bei-bei and Jin Xue-bo

School of Computer and Information Engineering,
Beijing Technology and Business University, Beijing, 100048, China
Tel.: 15810018803, 13691595989
Miaobeibei1@163.com, xuebojin@gmail.com

Abstract

Most time series big data is with noise and uncertain. To abstract the key information effectively and quickly, the estimation is one of the feasible methods for the uncertain big data. The Kalman filter with adaptive method by part of samples can give the high dimensional characteristics, reduce the computing cost and data uncertainty, but encounter the irregular estimation. The number of sample and the performance of the abstracted information have the tradeoff, which means we can use the suitable number of sample to abstract the key information of the series data. This paper discusses how to find the suitable sampling points for the time series data and the simulations show that the key dynamic information of time series big data can be guaranteed with the compression amount number of sample data.

Keywords: *Dynamic Guaranteed Cost compression; Time series big data; Kalman filter; estimation performance; estimation covariance*

1. Introduction

Among all the definitions offered for "big data", one of the popular expressiones is that it means data that is too big, too fast, and too hard to process. Here, "too big" means that organizations increasingly must deal with petabyte-scale collections of data that come from click streams, transaction histories, sensors, and elsewhere [1, 2]. "Too fast" means that not only the data is big, but it must be processed quickly, for example, to perform fraud detection at a point of sale or decide which ad to show to a user on a webpage. In other words, most big data is with time series relationship [3, 4]. "Too hard" is a catchall for data that doesn't fit neatly into one of existing processing tools or that needs some kind of analysis existing tools can't readily provide.

As we know, one reason of "hard to process" is the data with the noise and uncertainty in the most time especially from sensors, except the aforementioned "big" , then the "fast" processing of the data becomes more difficult. For the general data mining or classification method, the uncertainty is one of the main difficulties to get the reasonable results. Therefore for the research work, the big data is a challenging issue in the data sequence analysis area.

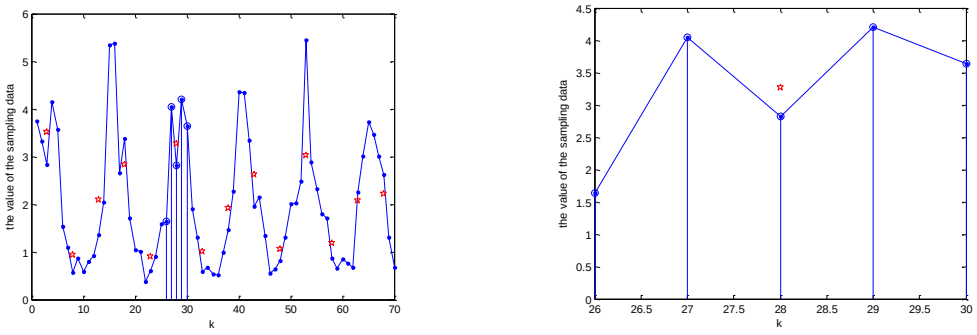
To get the useful information effectively and quickly is the key issue for the process of the big data. Especially in the field of cleaning noise as well as the compression of huge amounts time-series data. Take some Internet company for example, some of the operation and maintenance engineer tried to abstract part of monitoring data so as to detect abnormal points fast and accurately. Another application is the SF express company which forms a "multi-dynamic" type of Saving Algorithm to ensure both cost savings and timeliness when distribute express mails. [5]

Mostly, the first method we can use may be to cut dawn the "big" data in to a "small" time interval, and the mean of the interval is calculated to get one point, then the data amount is cut down. For example, the data chain with 70 sampling points shown in Fig.

ISSN: 2005-4270 IJDTA

1(a), we can cut dawn it into the time interval with 5 sampling points, i.e., [1,2,3,4], [6,7,8,9,10,11], etc. In each interval, we calculate the mean of total 5 data and get the mean to replace the former five points, which is shown by the "star" in the Figure 1.To show clearly, Figure 1 (b) gives one of the time intervals from 26 to 30. The "star" in the 28 with the mean value 3.27 is obtained and will use to replace the five data in the sampling point 26, 27, 28, 29, 30. That means the amount number of data is cut dawn to 1/5. While from the Figure 1, we can see this method can't remove the uncertainty of the time series data and the extracted data can't describe the original data effectively.

The estimation method is one of the effective methods for the uncertain and random signal [6, 7]. With the process model and measurement model, we can get the multi-state online estimation from the measurement by K to K+1 sampling. Almost all the classical estimation methods focus on the data with regular sampling time, which means the measurement is sampling and measured with the same interval, and the estimated state is obtained and more consistent with the true value. But for the series big data, the data chain is very long and the computing cost is too big if we use all the measurement data and it is really "hard" to get the "fast" process.



(a) (b)

Figure 1. The Mean Method for the Long Time Series Chain
(a) The Data Chain with the Sampling Time from 0 to 70
(b) The Data Chain with the Sampling Time from 26 to 30

Therefore the "fast" estimation method is deserved to be discussed. A method named as the irregular estimation method is given in Ref. [8] and Ref. [9] uses this method to implement the fast video tracking with less computing cost. Ref. [10] provides a solution with randomly selected measurement for the target tracking system, where the irregular sampling interval is transformed to a time-varying parameter by calculating the matrix exponential, and the dynamic parameter is estimated by the online estimated state with Yule-Walker method.

Ref. [11] discussed the convergence of the method in [10] and discussed relation of the sampling time, the number of the sampling and the performance of the estimation. The paper concluded that the reduction of sampling points results in the reduction of the measurement information, and causes the performance degrade.

By the Ref. [8-11], we can see the irregular estimation method with selected measurement has the following two advances: 1) the uncertainty is effectively reduced and the necessary information is extracted by the estimation processing. 2) the calculation cost is cut down by selecting some of the measurement. While as the Ref. [11] had noted that the estimation performance will degrade. Thus the relationship should be found between the sampling points and the performance of system is very important, and it is necessary to find a tradeoff between the computing speed and the estimating performance.

Comparing the tracking problem [11-14], this paper discusses the guaranteed cost compression for the series big data problem. The dynamic model based on the Newton's

law of kinematics is widely used in the maneuvering target tracking, while whether it is adaptive to the other series data is need to research. This paper pushes the former works and uses the irregular estimation to the series big data, and studies the performance estimation including the calculation time and the estimation covariance, by which found a tradeoff between the computing speed and estimation performance.

This paper is organized as follows: Section 2 gives the models under irregular sampling. Section 3 gives the estimation method based on Kalman filter. The simulations and experiments are provided in section 4. Finally, concluding remarks are given in Section 5.

2. The Adaptive Dynamic Model

In generally, the dynamic model is used for the target tracking problem. x, $\dot{x}$ and $\ddot{x}$ is described as the position, velocity, and acceleration of the target, respectively. Specifically, $\ddot{x}(t) = a(t)$. The state vector is always taken to be $x = [x, \dot{x}, \ddot{x}]'$ along the generic direction, unless stated otherwise explicitly. As to the other time series data chain, we can regard x as the real value, and the velocity, and acceleration comparing the measurement with noise.

Set $th_i = t_{i+k} - t_i$, where t_i is the sampling time of current measurement, and t_{i+k} is the sampling time of next measurement. We get the discrete-time equivalent as the following

$$x(t_{i+k}) = A_d(t_i)x(t_i) + w_d(t_i) \tag{1}$$

where

$$A_d(t_i) = \begin{bmatrix} 1 & th_i & {\alpha th_i - 1 + e^{-\alpha th_i}}/{\alpha^2} \\ 0 & 1 & {1 - e^{-\alpha th_i}}/{\alpha} \\ 0 & 0 & e^{-\alpha th_i} \end{bmatrix} \tag{2}$$

The covariance of the $w(k)$ as

$$Q_d(t_i) = E[w_d(t_i)w_d^T(t_i)] = 2\alpha\delta_\alpha^2 \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \tag{3}$$

$$\begin{aligned}
q_{11} &= \frac{1}{2\alpha^5}\left[1 - e^{-2\alpha \cdot th_i} + 2\alpha \cdot th_i + \frac{2\alpha^3 th_i^3}{3} - 2\alpha^2 th_i^2 - 4\alpha \cdot th_i e^{-\alpha \cdot th_i}\right] \\
q_{12} &= \frac{1}{2\alpha^4}\left[e^{-2\alpha \cdot th_i} + 1 - 2e^{-\alpha \cdot th_i} + 2\alpha \cdot th_i e^{-\alpha \cdot th_i} - 2\alpha \cdot th_i + \alpha^2 th_i^2\right] \\
q_{13} &= \frac{1}{2\alpha^3}\left[1 - e^{-2\alpha \cdot th_i} - 2\alpha \cdot th_i e^{-\alpha \cdot th_i}\right] \\
q_{22} &= \frac{1}{2\alpha^3}\left[4e^{-\alpha \cdot th_i} - 3 - e^{-2\alpha \cdot th_i} + 2\alpha \cdot th_i\right] \\
q_{23} &= \frac{1}{2\alpha^2}\left[e^{-2\alpha \cdot th_i} + 1 - 2\alpha \cdot th_i\right] \\
q_{33} &= \frac{1}{2\alpha}\left[1 - e^{-2\alpha \cdot th_i}\right]
\end{aligned} \tag{4}$$

where $x(t_i) = \left[x(t_i) \quad \dot{x}(t_i) \quad \ddot{x}(t_i)\right]'$, the parameter α is so called the reciprocal of the maneuver frequency for the target tracking. While for other time series data, α should be decided by the changing characteristics of the data. Then we rewrite the discrete state–space model of the tracking system as

$$\begin{aligned}
x(t_{i+k}) &= A_d(t_i)x(t_i) + w_d(t_i) \\
z(t_i) &= H(t_i)x(t_i) + v(t_i)
\end{aligned} \tag{5}$$

where $x(k) = \left[x(k) \quad \dot{x}(k) \quad \ddot{x}(k)\right]'$ is the state of the system to be estimated and whose initial

mean and covariance are known as x_0 and P_0, $w_d(t_i)$ and $v(t_i)$ are white noise with zero mean and independent of the initial state x_0, $z(t_i)$ is the measurement vector, $H(t_i)$ is measurement matrices and $v(t_i)$ is measurement noise with known variance *R*. We can see the same sampling interval is just a particular case of the random sampling problem. Therefore the model of the randomly sampling tracking is a general one.

How to decide the maneuver frequency α and the variance of the acceleration δ_a^2 in (2) and (3) is called as modeling the dynamic characteristic. Unlike the Singer model and current model, we assume α and δ_a^2 are not constant but variable and be expressed as α_i and $\delta_{a\,i}^2$. From the processing model of (5), we have the discrete time equation of the acceleration as

$$a(t_{i+k}) = \beta_i a(t_i) + w^a(t_i) \tag{6}$$

where

$$\beta_i = e^{-\alpha_i th_i} \tag{7}$$

and $w^a(t_{i-1})$ is a zero-mean white noise sequence with the variance

$$\delta_{aw\,i}^2 = \delta_{a\,i}^2(1-\beta_i^2) \tag{8}$$

For a first-order stationary Markov process (6), we can get the parameter β_i and δ_{awi}^2 by the statistics relation between the autocorrelation function $r(0)$, $r(1)$ of $a(t_i)$ with by the Yule-Walker method [16]. Next we can get α_i and $\delta_{a\,i}^2$ by $\delta_{a\,i}^2 = \delta_{awi}^2 \big/ 1-\beta_i^2$, $\alpha_i = \ln \beta_i \big/ -th_i$, then get the system parameters $A_d(t_i)$ and $Q_d(t_i)$ in process function (2)-(3).

3. The Estimation Method for the Series Big Data

The following is the algorithm in the predictor–corrector form by Kalman filter, which is convenient for implementation:

1) Initialization: $i = 0$

$$\hat{x}(t_0 \mid t_0) = x_0, P(t_0 \mid t_0) = P_0,\ \alpha_0,\ \delta_{a0}^2,\ r_0(t_0) = \ddot{x}_0 \cdot \ddot{x}_0,\ r_0(t_1) = \ddot{x}_0 \tag{9}$$

2) Recursion: $i := i + k$

a) We first create a uniform distribution random number Sa with the range from 0 to 1, then the following algorithm is used to choose the next measurement to be processed. Set $k = 0$ and a positive constant A, where $A \in (0,1)$.

If $Sa < A$, then $k \leftarrow k+1$

If $Sa > A$, then the next measurement is picked with an integer k.

b) System update:

Set $th_i = t_{i+k} - t_i$, and the system parameter as

$$\hat{A}_d(t_i) = \begin{bmatrix} 1 & th_i & \dfrac{\alpha_i th_i - 1 + e^{-\alpha_i th_i}}{\alpha_i^2} \\ 0 & 1 & \dfrac{1-e^{-\alpha_i th_i}}{\alpha_i} \\ 0 & 0 & e^{-\alpha_i th_i} \end{bmatrix} \tag{10}$$

and the variance of the $w_d(t_i)$ as

$$\hat{Q}_d(t_i) = E[w_d(t_i)w_d^T(t_i)] = 2\alpha_i\delta_{\alpha\,i}^2 \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \tag{11}$$

with parameters described as

$$\begin{aligned} q_{11} &= \frac{1}{2\alpha_i^5}\left[1 - e^{-2\alpha_i \cdot th_i} + 2\alpha_i \cdot th_i + \frac{2\alpha_i^3 th_i^3}{3} - 2\alpha_i^2 th_i^2 - 4\alpha_i \cdot th_i e^{-\alpha \cdot th_i}\right] \\ q_{12} &= \frac{1}{2\alpha_i^4}\left[e^{-2\alpha_i \cdot th_i} + 1 - 2e^{-\alpha_i \cdot th_i} + 2\alpha_i \cdot th_i e^{-\alpha_i \cdot th_i} - 2\alpha_i \cdot th_i + \alpha_i^2 th_i^2\right] \\ q_{13} &= \frac{1}{2\alpha_i^3}\left[1 - e^{-2\alpha_i \cdot th_i} - 2\alpha_i \cdot th_i e^{-\alpha_i \cdot th_i}\right] \\ q_{22} &= \frac{1}{2\alpha_i^3}\left[4e^{-\alpha_i \cdot th_i} - 3 - e^{-2\alpha_i \cdot th_i} + 2\alpha_i \cdot th_i\right] \\ q_{23} &= \frac{1}{2\alpha_i^2}\left[e^{-2\alpha_i \cdot th_i} + 1 - 2\alpha_i \cdot th_i\right] \\ q_{33} &= \frac{1}{2\alpha_i}\left[1 - e^{-2\alpha_i \cdot th_i}\right] \end{aligned} \tag{12}$$

c) State prediction:

$$\hat{x}(t_{i+k} \mid t_i) = \hat{A}_d(t_i)\hat{x}(t_i \mid t_i) \tag{13}$$

$$P(t_{i+k} \mid t_i) = \hat{A}_d(t_i)P(t_i \mid t_i)\hat{A}_d^T(t_i) + \hat{Q}_d(t_i) \tag{14}$$

d) State updation:

$$\hat{x}(t_{i+k} \mid t_{i+k}) = \hat{x}(t_{i+k} \mid t_i) + K(t_{i+k})[z(t_{i+k}) - H(t_{i+k})\hat{x}(t_{i+k} \mid t_i)] \tag{15}$$

$$K(t_{i+k}) = P(t_{i+k} \mid t_i)H^T(t_{i+k})[H(t_{i+k})P(t_{i+k} \mid t_i)H^T(t_{i+k}) + R(t_{i+k})]^{-1} \tag{16}$$

$$P(t_{i+k} \mid t_{i+k}) = [I - K(t_{i+k})H(t_{i+k})]P(t_{i+k} \mid t_i) \tag{17}$$

e) Parameter Adaptation:

When $i \le K_0$, the maneuver frequency α_i is set to α_0 and the covariance of the noise $\delta_{a\,i}^2$ is set to $\delta_{a\,0}^2$, which are get by the following

$$\alpha_0 = 1/20\ ,\ \delta_{\alpha\,0}^2 = 25 \times (4 - \pi)/\pi \tag{18}$$

When $i > K_0$, the parameter is updated by the following

$$\hat{a}(t_{i+k}) = \hat{\dot{x}}(t_{i+k} \mid t_{i+k}) \tag{19}$$

$$r_{i+k}(1) = r_i(1) + \frac{1}{i}\left[\hat{a}(t_{i+k})\hat{a}(t_i) - r_i(1)\right] \tag{20}$$

$$r_{i+k}(0) = r_i(0) + \frac{1}{i}\left[\hat{a}(t_{i+k})\hat{a}(t_{i+k}) - r_i(0)\right] \tag{21}$$

$$\beta_{i+k} = \frac{r_{i+k}(1)}{r_{i+k}(0)} \qquad \delta_{aw\ i+k}^2 = r_{i+k}(0) - \beta_{i+k}r_{i+k}(1) \tag{22}$$

$$\delta_{a\ i+k}^2 = \frac{\delta_{aw\ i+k}^2}{1 - \beta_{i+k}^2} \qquad \alpha_i = \frac{\ln \beta_{i+k}}{-th_{i+k}} \tag{23}$$

then use Eq. (10) - (12) to get the system parameter $A_d(t_i)$ and $Q_d(t_i)$.

Note1: $r_k(0)$ and $r_k(1)$ are the autocorrelation functions, which need to have statistical data. In the simulation, we set the two parameters change after 4-20 steps. Here we set $k_0 = 4$.

Based on the closed-loop estimation algorithm [10], we can see the parameter used to estimate state is an estimated one and similarly, the estimated states to calculate the parameters α_i and $\delta_{a\,i}^2$ have estimation errors, too. So it's important to guarantee the convergence of the estimation of the states and parameters. From paper [9], we can see if one step predictive covariance is bounded, i.e., $P(k \mid k-1) \leq P_0$, $P(k+1 \mid k)$ must be bounded with the fact $\hat{Q}_d(k) \leq Q_0$. And $K(k+1)$ must be a bounded matrix and $\hat{x}(k+1 \mid k+1)$ must be bounded, too.

Note3: we set δ_a^2 a bounded value 10000 to avoid system divergence.

Note4: We update the model parameters α and δ_a when $\beta > 0$ only, because if $\beta < 0$, we can't get a reasonable α_i by $\alpha_i = \frac{\ln \beta_{i+k}}{-th_{i+k}}$ and α_i is obtained as a complex number, which is an improper case for the problem discussed here.

4. Simulation Results

Based on the algorithm developed here we will analyze the relation between the estimation performance and the number of selected sampling points, and define the following Compression Sampling Rate (CSR)

$$CSR = \frac{\text{the number of selected sampling points}}{\text{the total number of the sample}} \tag{19}$$

We can see lower CSR means a higher compression rate.

Two cases are discussed in the following: Case 1 is the classical tracking problem. At a simulation planar, a target with long trajectory is traced and tracked quickly with suitable number of the sampling data. Case 2 is to study the financial time series data by the developed estimation method, the fast estimation is executed and the data is compressed with the guaranteed cost.

Case 1.

The proposed algorithm is applied to a two dimensional planar tracking problem to verify the performance through a series of simulation runs. The measurement noises are generated as white Gaussian random numbers with variance *R*. We assume *R* is a given parameter, as $R = 25$ .The initial state estimate x_0 and covariance P_0 are assumed to be $x_0 = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]'$ and $P_0 = diag(10,10,10,10,10,10)$. The actual trajectory of the target (contents 3221points) is shown in Figure 2. Due to the measurement noise we can only get the trajectory sequence with noise shown in Figure 3, where the "star" is described the each sampling measurement, and the solid line is the actual trajectory.

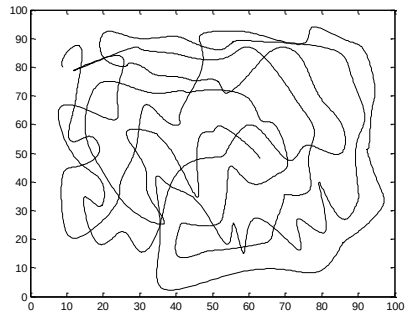


Figure 2. The Actual Trajectory of the Target

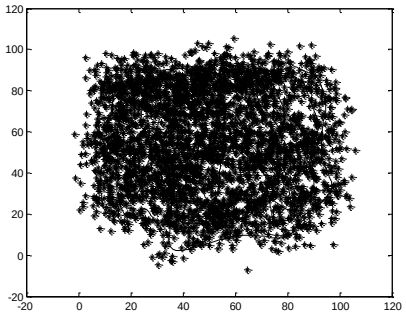


Figure 3. The Measurement of the Target

By changing the parameter of *A* we can get different CSR results. *A* is set between 0-1. Larger *A* means fewer points, while smaller *A* means more points. Based on the developed algorithm about guaranteed cost compression method, we give the estimation results about different A with 0.2, 0.4, 0.6, 0.8, i.e., CSR with 79.8%, 61.6%, 40.7%, 18.9% for a sample with 3221 points.

To better illustrate our results, we give the comparison of the real and the estimates trajectories. The estimations of horizontal and longitudinal axis are shown in Figure 4 and Figure 5, respectively. The location estimation errors show in Figure 6 and Figure 7, respectively. We can see when *A* is a less value, the covariance will be larger, while the CSR is higher and the selected sampling points is less.

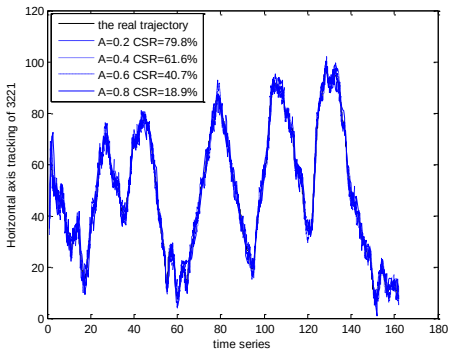


Figure 4 Different CSR horizontal estimates of trajectory with 3221 points

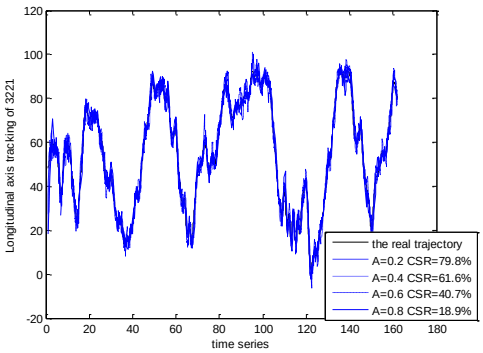


Figure 5 Different CSR longitudinal estimates trajectory with 3221 points

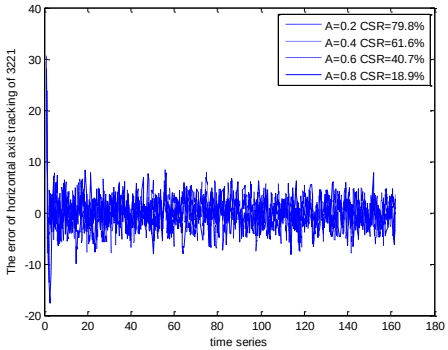


Figure 6. Different CSR of the horizontal estimate error of trajectory with 3221 points

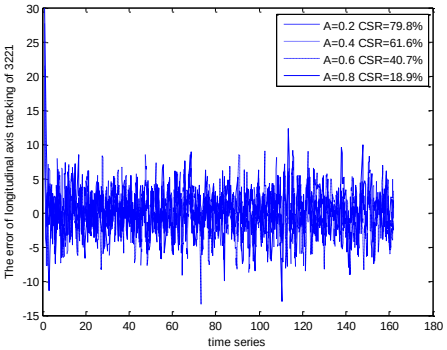


Figure 7. Different CSR of the longitudinal estimate error of trajectory with 3221 points

To illustrate how the compression sampling rate affects estimation performance, we choose the estimated covariance as a metric.

$$\begin{aligned} \hat{x}(i) - x(i) &= e_x(i) \\ \hat{y}(i) - y(i) &= e_y(i) \\ COV &= \sum_{i=1}^{N} e^2(i) \Big/ N \\ e^2(i) &= e^2{}_x(i) + e^2{}_y(i) \end{aligned} \tag{20}$$

Table 1 gives some of the estimated covariance under different compression sampling rate (CSR) for different trajectory and Figure 8 gives the graph representation of Table 1.

Table 1. Relation between Compression Sampling Rate and Estimated Covariance

Estimated Covariance		CSR										
		11.50%	20.80%	29.96%	39.50%	49.82%	59.10%	67.10%	70.91%	81.69%	87.10%	100%
The trajectory with long chain	681 points	55.050	49.927	30.120	30.957	33.280	23.611	22.316	24.000	20.882	18.388	16.474
	961 points	39.221	31.114	24.508	23.014	24.304	17.327	14.642	16.510	15.675	19.223	11.603
	1491 points	35.246	26.237	34.237	14.984	12.530	19.475	19.163	16.454	11.894	24.259	8.528
	1801 points	21.744	59.510	42.379	33.870	24.069	22.937	18.071	19.546	15.684	14.861	13.534
	2001 points	23.395	21.078	17.459	12.296	15.273	10.114	8.332	10.481	7.519	10.768	8.166
	2401 points	18.563	18.563	15.691	12.283	11.616	12.187	8.889	7.805	7.675	11.845	12.392
	2991 points	37.773	30.932	25.044	18.807	16.890	15.127	15.116	14.600	12.740	13.120	11.615
	3221 points	33.656	26.557	40.435	16.025	18.523	11.079	12.373	10.824	11.226	10.407	9.437
	3601 points	31.421	26.691	23.319	19.443	21.539	17.420	16.038	14.807	12.092	11.533	10.494
	4161 points	23.270	15.040	18.726	10.897	19.695	15.256	11.767	15.133	10.549	9.934	8.403
	4601 points	36.265	25.756	22.357	26.463	18.336	15.815	15.976	16.582	13.855	12.488	11.144

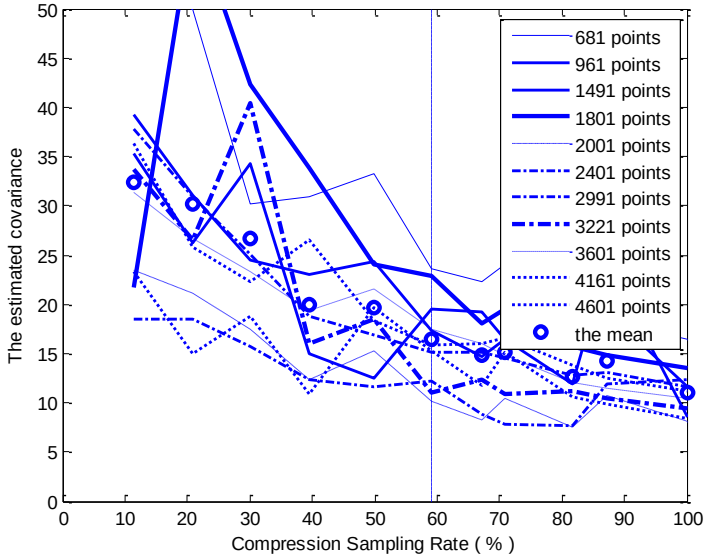


Figure 8. The Estimation Covariance of the Location of Sampling Points of different Trajectory with Long Chain

Figure 8 also gives the mean value of estimation covariance of the different trajectory by the "circle". We can see the sampling rate doesn't have much impact on the estimated covariance especially larger than 59.10% CSR (this point is shown in the Figure 7 by the vertical dashed line), so we can conclude that there exist the tradeoff between the computing speed and the estimating performance when CSR is about 59.10%. The lower sampling rate means the less measured data is get, and the less useful information can be provided, but the estimation process is faster.

Case2.

To illustrate the developed method has the wide applicability, not only the classical tracking problem, but the other time series big data, we then use the developed method to extract the useful information of the finance forex data quickly. We use the opening price data of the AUD/CAD exchange rate from 0:00 August fifth 2013 to 16:25 August ninth 2013, every five minutes comes one data (the data set contains 1350 in total). By the developed compression method, the estimation covariance of the forex data is shown in

Table 2 with the different CSR. Figure 9 gives the graph representation of the Table 2. We can see with the CSR from 11.19% to 100%, the different estimation covariance is obtained. Table 2 shows that CRS hasn't developed the estimated covariance very much from 60%, which means if the estimation about 0.12 is enough, the computing can be cut down to 60% of the whole computing cost.

We then use the calculating mean value method shown in Figure 1 to consider the mean value of every six data of the forex data and get the covariance 0.091. That means the compute cost decreased to about one sixth comparing the total computing cost (the CSR is about 16.66%). While we can see in the Table 2, with the fewer computing cost (for example CSR=11.19%), the developed method can get the covariance 0.040. Therefore in terms of the guaranteed cost compression, the developed method here can get the better estimation performance with lower computing cost.

Table 2. Relation between Compression Sampling Rate and the Estimated Covariance of Forex Data

CSR	100%	90.96%	79.03%	69.48%	60%	50.52%	41.10%	30.30%	20.20%	11.19%
cov	0.009	0.010	0.011	0.011	0.012	0.014	0.015	0.018	0.024	0.040

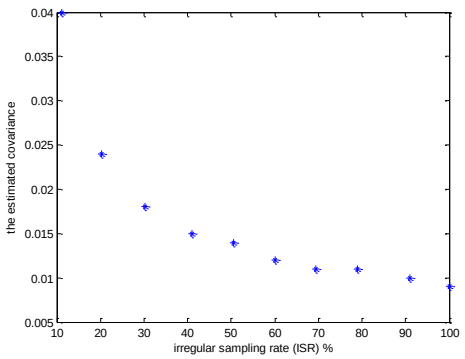


Figure 9. The Estimation Covariance of AUD/CAD Exchange Rate

5. Conclusions

The main contribution of this paper is to give a method to randomly select the next measurement and analysis how the randomly selected measurements and compression sampling rate will effect the estimation performance. We found that by the developed guaranteed cost compression method, only some measurements needed and computing cost is reduced greatly. But we can also see with the larger CSR lead to the worse estimation performance because the less useful information is used. Thus we conclude that a tradeoff between the computing speed and the estimating performance is existed. The future work will focus on how to find the tradeoff point based on the series big data, not only by the estimation covariance, but by the characteristic of the measurement.

What we like to discuss further is that though the amount of the data used in this paper is not too "big", only have several thousands of number. While the cut-down-amount is shown by percentage, which means if we can cut down the forex data with 1350 points to 60%, we also can cut down the data amount to 60% when we process the "real" big data with petabyte-scale.

Acknowledgements

This work is partially supported by NSFC under Grant No. 61273002, 60971119 and the Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions No. CIT&TCD201304025.

References

[1] Z. Chan, "Research of big-data mining visualization application", 13th International Conference on Man-Machine-Environment System Engineering, **(2014)**; Springer Verlag.
[2] W. Xindong, Z. Xingquan, W. Gong-Qing and D. Wei, "Data mining with big data", IEEE Transactions on Knowledge and Data Engineering, vol.26, n.1, **(2014)**, pp.97-107.
[3] Sengupta, Neha, Aloka, S. Narayanaswamy, Balakrishnan, Ismail, Hamidah, Mathew and Satyajith, "Time series data mining for demand side decision support", 2013 IEEE Innovative Smart Grid Technologies - Asia, ISGT Asia **(2013)**.
[4] L. Shen, Maharaj, E. Ann, Inder and Brett, Polarization of forecast densities: A new approach to time series classification. Computational Statistics and Data Analysis, vol.70, **(2014)**, pp. 345-361.
[5] L. Junchao, W. Yuhong, "Express company's vehicle routing optimization by multiple-dynamic saving algorithm", Journal of Industrial Engineering and Management, vol.7, **(2014)**, pp.390-400.
[6] M. Nicholas and G. Dimitry, "Covariance estimation in two-level regression", 2nd International Conference on Control and Fault-Tolerant Systems, **(2013)**.
[7] N. Ahmed O., T. Zhi and L. Qing, "High-dimensional sparse covariance estimation for random signals. 2013 38th IEEE International Conference on Acoustics, Speech, and Signal Processing, **(2013)**.
[8] J. Xue-bo, D. Jing-jing and B. Jia, "Target Tracking of a Linear Time Invariant System Under Irregular Sampling", International Journal of Advanced Robotic Systems, vol.9, **(2012)**, pp. 1-12.
[9] J.Xue-Bo, D. Jing-Jing and B. Jia, "Fast tracking for video target tracking", Applied Mechanics and Materials, vol.303-306, **(2013)**, pp. 2245-2248.
[10] J. Xue-bo, L. Xiao-feng, S. Ting-li, S. Yan and M. Bei-bei, "Closed-Loop Estimation for Randomly Sampled Measurements in Target Tracking System", Mathematical Problems in Engineering, vol.2014, (**2014**).
[11] M. Beibei and J. Xuebo, "Relation between Irregular Sampling and Estimated Covariance for Closed-loop Tracking Method", The14th International Conference on Algorithms and Arcgitectures for Parallel Processing, vol.8630, **(2014)**, pp.836-843.
[12] Y. Jiang and J. Xiao, "Target tracking based on a multi-sensor covariance intersection fusion kalman filter", Engineering Review, vol.34, **(2014)**, pp. 47-54.
[13] J. A. Said and Y. Suet-Peng, "Unscented Kalman filter for noisy multivariate financial time-series data", In: 7th Multi-Disciplinary International Workshop on Artificial Intelligence LNAI, **(2013)**, Springer Verlag.
[14] Z. Haitao, D. Gang, S. Junxin and Z. Yujiao, "Unscented Kalman filter and its nonlinear application for tracking a moving target", Optik, vol.124, **(2013)**, pp. 4468-4471.
[15] S. V. Bordonaro, W. Peter and B.-S. Yaakov, "Performance analysis of the converted range rate and position linear Kalman filter", The 47th Asilomar Conference on Signals, Systems and Computers **(2013)**.
[16] H. H. Zhang, M. V. Basin and M. Skliar, "Ito-Volterra optimal state estimation with continuous, multirate, randomly sampled, and delayed measurements", IEEE Transactions on Automatic Control, (**2007**).

为方便组委会联系，请提供 2 位作者信息。

论文题目	**Dynamic Guaranteed Cost Compression for Time Series Big Data**		
所属主题	**Compression Methodology**		
第一作者			
姓名	苗贝贝	职称/学位	研究生在读
单位	北京工商大学	邮编	100048
地址	北京市海淀区阜成路 11 号		
电话		手机	18810926092
Email	Miaobeibei1@163.com		
第二作者			
姓名	金学波	职称/学位	教授
单位	北京工商大学	邮编	100048
地址	北京市海淀区阜成路 11 号		
电话		手机	13691595989
Email	xuebojin@gmail.com		