Preprints of the 18th IFAC World Congress
Milano (Italy) August 28 - September 2, 2011

Active Learning of Gaussian Processes for Spatial Functions in Mobile Sensor Networks

Dongbing Gu * **Huosheng Hu** *

* *University of Essex, Wivenhoe Park, CO4 3SQ UK (e-mail: dgu@essex.ac.uk, hhu@essex.ac.uk).*

Abstract: This paper proposes a spatial function modeling approach using mobile sensor networks, which potentially can be used for environmental surveillance applications. The mobile sensor nodes are able to sample the point observations of an $2D$ spatial function. On the one hand, they will use the observations to generate a predictive model of the spatial function. On the other hand, they will make collective motion decisions to move into the regions where high uncertainties of the predictive model exist. In the end, an accurate predictive model is obtained in the sensor network and all the mobile sensor nodes are distributed in the environment with an optimized pattern. Gaussian process regression is selected as the modeling technique in the proposed approach. The hyperparameters of Gaussian process model are learned online to improve the accuracy of the predictive model. The collective motion control of mobile sensor nodes is based on a locational optimization algorithm, which utilizes an information entropy of the predicted Gaussian process to explore the environment and reduce the uncertainty of predictive model. Simulation results are provided to show the performance of the proposed approach.

Keywords: Coverage control, Gaussian process regression, Active sensing, Mobile sensor networks, Spatial function modeling.

1. INTRODUCTION

The environmental surveillance in meteorology and climatology, ecology, demography, epidemiology, forestry, fisher, oceanography, and others requires the capability of modeling spatial functions, or even spatial-temporal functions. The distributed nature in spatial space and the mobile capability in temporal space of mobile sensor networks offer an capability for environmental surveillance applications. Several research projects have targeted to this research area, such as monitoring forest fires using UAVs in Merino et al. (2006), monitoring air quality using UAVs in Corrigan et al. (2007), monitoring ocean ecology conditions using UWVs in Leonard et al. (2007).

Mobile sensor networks are able to make sensing observations of environmental spatial function with their on-board sensors, exchange information with on-board wireless communication, and explore the environment with their mobility. Consequently they are able to produce a predictive model based on sensing observations and allocate themselves in a pattern which can generate a more accurate predictive model. Gaussian process (GP), also known as Kriging filter, is a well-known regression technique for data assimilation. GPs are specified by a mean function, a covariance function, and a set of hyperparameters which can be determined from a training set. The learning algorithm of hyperparameters is based on maximizing the marginal likelihood. The main advantage of GP regression over other regression techniques is the ability of predicting not only the mean function, but also the covariance function, see Williams and Rasmussen (1996), MacKay (1998), Rasmussen and Williams (2006).

The predictive uncertainty is valued information for further decision making in environmental surveillance applications. Recent publications using GPs to model a spatial function include Krause et al. (2008), Stranders et al. (2008), Stachniss et al. (2009), Ny and Pappas (2009), Cortes (2009), Singh et al. (2010). In Krause et al. (2008), GP regression was applied for monitoring the ecological condition of a river. The sensor placement was determined by maximizing a mutual information gain, which selects locations which most effectively reduce the uncertainty at the unobserved locations. GP regression is a non-parameter regression technique and its computation complexity will grow with the size of sampled data. In Stranders et al. (2008), the computation complexity of GP regression was reduced by a Bayesian Monte Carlo approach, and an information entropy was used to allocate mobile sensor nodes. In Stachniss et al. (2009), a mixture of GPs was applied for building a gas distribution with the aim to reduce the computation complexity. In Ny and Pappas (2009), a Kalman filter was built on the top of a GP model to characterize spatial-temporal functions. A path planning problem was solved by optimizing a mutual information gain via an effective computation algorithm. In Cortes (2009), a Kriged Kalman filter was developed to build spatial-temporal functions. A centroidal Voronoi tessellation (CVT) algorithm was employed to allocate mobile sensor nodes according to the predictive spatial function. The Kriged Kalman filter and swarm control were developed in Choi et al. (2008) to build spatial-temporal functions. In Singh et al. (2010), several non-separable spatial-temporal covariance functions were proposed for modeling spatial-temporal functions. The same mutual information gain as in Krause et al. (2008) was utilized to plan the path for mobile sensor nodes.

Environmental spatial functions have been modeled in mobile sensor networks by using RBF networks in Schwager et al. (2009) where a CVT coverage control was used to allocate mobile sensor nodes, and in Lynch et al. (2008) where a flocking control was used to move sensor nodes. RBF is a truncated GP regression where limited number of base functions is used. Environmental spatial function was approximated by using an inverse distance weighting interpolation method and updated by using a Kalman filter in Martinez (2010). A Kalman filter approach for dynamic coverage control was also proposed in Hussein (2007).

In this paper, we propose to use GP regression to build a spatial function with a mobile sensor network. Our contribution is to make the hyperparameters of GP model adaptive online so that a more accurate time varying predictive model can be obtained. An information entropy of the predictive Gaussian process is optimized to allocate mobile sensor nodes so that the environment can be explored and the model uncertainty can be reduced. A CVT algorithm that uses the information entropy as a utility function is proposed in this paper. This algorithm is able to allow mobile sensor nodes to make collective motion decisions and allocate themselves in a pattern which can reduce the uncertainty of the predictive GP model.

In the following, Section 2 presents the basics of Gaussian process regression and the hyperparameter learning algorithm. The information entropy based coverage control and its integration with the CVT algorithm are introduced in Section 3. Section 4 provides simulation results. Our conclusion and future work are given in Section 5.

2. GAUSSIAN PROCESS REGRESSION

A mobile wireless sensor network with N sensors is to be deployed in an $2D$ area Q to model a scalar environmental spatial function in that area. Sensor node i is located at a $2D$ position $\mathrm{x}_{i,t}$ and it is assumed that the position $\mathrm{x}_{i,t}$ can be found by itself with self-localization techniques at time step t. Each sensor node i can make a point observation $y_{i,t}$ of an environmental spatial function $f(\mathrm{x}_{i,t})$ at time step t. The sensory observation distribution is assumed to be Gaussian:

$$y_{i,t} = f(\mathrm{x}_{i,t}) + \varepsilon_{i,t}$$

where $\varepsilon_{i,t}$ is a Gaussian noise with mean zero and covariance σ_t^2 noted as $\varepsilon_{i,t} \sim \mathcal{N}(0, \sigma_t^2)$.

It is assumed that each sensor node can collect all the location information $\mathrm{x}_{j,t}$ and its corresponding observation $y_{j,t}$ from all the other sensor nodes via wireless communication.

2.1 Gaussian Process

In a sensor node, Gaussian inference is conducted at each time step based on the given information available at that moment. The given information includes a data set $\mathcal{D}_t$ of input vectors $X_t = [\mathrm{x}_{1,t}, \dots \mathrm{x}_{N,t}]^T$ and the corresponding observations $\mathrm{y}_t = [y_{1,t}, \dots, y_{N,t}]^T$.

In a GP model, the prior distribution of latent variable $f_{i,t} = f(\mathrm{x}_{i,t})$ is modeled as Gaussian. Its mean value is assumed to be zero because offsets and simple trends can be subtracted out first. The prior knowledge about multiple latent variables is modeled by a covariance function $K_{NN,t} = [k(\mathrm{x}_{i,t}, \mathrm{x}_{j,t})]$. With a positive definite covariance function $K_{NN,t}$, the GP prior distribution of latent vector $\mathrm{f}_t = [f_{1,t}, \dots, f_{N,t}]^T$ is represented as:

$$p(\mathrm{f}_t) = \mathcal{N}(0, K_{NN,t})$$

The likelihood distribution of observation vector y_t is represented as:

$$p(\mathrm{y}_t|\mathrm{f}_t) = \mathcal{N}(\mathrm{f}_t, \sigma_t^2 I)$$

GP regression can infer $f_{*,t} = f(\mathrm{x}_{*,t})$ for a test point $\mathrm{x}_{*,t} \in Q$ using $p(f_{*,t}|\mathrm{y}_t)$ given a training data set (X_t, y_t) and a single test point $\mathrm{x}_{*,t}$. The latent predictive distribution of the given test point is obtained by solving the MAP problem and is given below:

$$p(f_{*,t}|\mathrm{y}_t) = \mathcal{N}(\mu_{*,t}, \Sigma_{*,t})$$

in which the predictive mean function and the predictive covariance function are:

$$\begin{aligned} \mu_{*,t} &= K_{*N,t}(K_{NN,t} + \sigma_t^2 I)^{-1}\mathrm{y}_t \\ \Sigma_{*,t} &= K_{**,t} - K_{*N,t}(K_{NN,t} + \sigma_t^2 I)^{-1} K_{N*,t} \end{aligned} \tag{1}$$

where $K_{N*,t} = K_{*N,t}^T$ for symmetrical covariance functions, and

$$\begin{aligned} K_{**,t} &= k(\mathrm{x}_{*,t}, \mathrm{x}_{*,t}) \\ K_{*N,t} &= [k(\mathrm{x}_{*,t}, \mathrm{x}_{1,t}), \dots, k(\mathrm{x}_{*,t}, \mathrm{x}_{N,t})] \end{aligned}$$

2.2 Hyperparameter Learning

The prior knowledge about $K_{NN,t}$ is very important for GP regression. It determines the properties of sample functions drawn from the GP prior and represents the prior knowledge about environmental spatial functions. For the mobile sensor network discussed in this research, the prior knowledge we have is that $f(\mathrm{x}_{i,t})$ is closely related to $f(\mathrm{x}_{j,t})$, i.e. $k(\mathrm{x}_{i,t}, \mathrm{x}_{j,t})$ approximates its maximum value if the distance between two nodes $\mathrm{x}_{i,t}$ and $\mathrm{x}_{j,t}$ is short. In contrast, $f(\mathrm{x}_{i,t})$ is not related to $f(\mathrm{x}_{j,t})$, i.e. $k(\mathrm{x}_{i,t}, \mathrm{x}_{j,t})$ approximates to zero if the distance between them is too far away. A valid covariance function guarantees that covariance matrix $K_{NN,t}$ is symmetrical positive definite. The commonly used covariance function is the 'squared exponential':

$$k(r_t) = a_t^2 \exp\left(-\frac{r_t^2}{l_t^2}\right) \tag{2}$$

where r_t is the Euclidean distance $||x_{i,t} - x_{j,t}||$ between two nodes. a_t is the amplitude and l_t is the lengthscale, both of which represent the characteristics of covariance function and are the hyperparameters of squared exponential covariance function. Notice that a smaller lengthscale implies the sample function varies more rapidly and a larger lengthscale implies the sample function varies more slowly. The hyperparameter set is denoted as $\theta_t = [a_t, l_t, \sigma_t]^T$. The time varying property of the hyperparameters is maintained via the maximum likelihood learning algorithm discussed below. Although temporal dynamics is not modeled in the GP regression discussed above, the time varying hyperparameters can potentially compensate for temporal dynamics.

Given a hyperparameter set, the log marginal likelihood is:

$$\begin{aligned}\mathcal{L} &= -\log p(\mathrm{y}_t|\theta_t) \\ &= \frac{1}{2}\mathrm{y}_t^T C_t^{-1}\mathrm{y}_t + \frac{1}{2}\log|C_t| + \frac{N}{2}\log(2\pi)\end{aligned}$$

where $C_t = K_{NN,t} + \sigma_t^2 I$. The partial derivative is:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \theta_t} &= -\frac{1}{2}\mathrm{y}_t^T C_t^{-1}\frac{\partial C_t}{\partial \theta_t}C_t^{-1}\mathrm{y}_t + \frac{1}{2}tr\left(C_t^{-1}\frac{\partial C_t}{\partial \theta_t}\right) \\ &= -\frac{1}{2}tr\left(\left(\alpha_t\alpha_t^T - C_t^{-1}\right)\frac{\partial C_t}{\partial \theta_t}\right) \end{aligned} \quad (3)$$

where $\alpha_t = C_t^{-1}\mathrm{y}_t$. From (2), it can find that

$$\begin{aligned}\frac{\partial C_t}{\partial a_t} &= \left[\frac{\partial k(r_t)}{\partial a_t}\right] = \left[2a_t\exp\left(-\frac{r_t^2}{l_t^2}\right)\right] \\ \frac{\partial C_t}{\partial l_t} &= \left[\frac{\partial k(r_t)}{\partial l_t}\right] = \left[2\frac{a_t^2 r_t^2}{l_t^3}\exp\left(-\frac{r_t^2}{l_t^2}\right)\right] \\ \frac{\partial C_t}{\partial \sigma_t} &= 2\sigma_t I\end{aligned}$$

3. INFORMATION ENTROPY BASED COVERAGE CONTROL

When the sensor network is deployed in the environment to be explored, they have no knowledge about the spatial function except the prior knowledge of GP regression. Mobile sensor node i makes observations $y_{i,t}$ at its location $\mathrm{x}_{i,t}$, and obtains observation $y_{j,t}$ and location $\mathrm{x}_{j,t}$ from all the other nodes $j \in N$ via wireless communication. It then learns the hyperparameters θ_t and constructs the covariance function $K_{NN,t}$. With the GP regression, a predictive mean function $\mu_{*,t}$ and a predictive covariance function $\Sigma_{*,t}$ are deduced. This is the first step of our proposed approach.

In the second step of our proposed approach, the sensor nodes need a strategy to explore the environment so that they will be able to model the spatial function more accurately. The predictive mean function $\mu_{*,t}$ could be used for generating a control signal $u_{i,t}$ for sensor node i. This strategy would lead to an allocation concentration of sensor nodes on the region where high mean values are available. However, this behavior can not reduce the uncertainty of GP regression. In this work, the information entropy is utilized for mobile sensor nodes to explore the environment in order to reduce the uncertainty of the predictive function. By optimizing the information entropy with respect to sensor node location $\mathrm{x}_{i,t}$, it is able to find a control input signal $\mathrm{u}_{i,t}$ for mobile sensor node i. Then sensor node i is able to move to the next position according to its kinematics $\mathrm{x}_{i,t+1} = \mathrm{x}_{i,t} + \mathrm{u}_{i,t}$.

3.1 Information Entropy

To reduce the uncertainty of GP regression, the predictive covariance function $\Sigma_{*,t}$ is a valued resource to be utilized. This evokes the use of posterior information entropy $H(f_{*,t}|\mathrm{y}_t)$, which measures the information potentially gained by making an observation.

The information entropy of Gaussian random variable $f_{*,t}$ conditioned on observation y_t is a monotonic function of its variance:

$$\begin{aligned}H(f_{*,t}|\mathrm{y}_t) &= \frac{1}{2}\log\left(2\pi e|\Sigma_{*,t}|\right) \\ &= \frac{1}{2}\log\left(2\pi e|K_{**,t} - K_{*N,t}C_t^{-1}K_{N*,t}|\right)\end{aligned}$$

As can be seen from the above equation, although it looks like that an observation will not be related to the information entropy, the observation affects the hyperparameters and therefore it does affect the covariance function. Thus this information entropy is dependent on actual observation. We can represent this dependence as follows:

$$H(\mathrm{x}_{*,t}) = H(f_{*,t}|\mathrm{y}_t)$$

3.2 Centroidal Voronoi Tessellation

According to the above discussion, optimizing the information entropy would be able to allow the mobile sensor nodes to explore the environment in order to reduce the uncertainty of GP regression. However, for multiple sensor nodes, when all of them optimize a single object function, it would be possible for them to move together rather than spreading over a large area to explore more high uncertain regions.

It is necessary for multiple mobile sensor nodes to cooperate when they explore the environment . The centroidal Voronoi tessellation (CVT) approach proposed in Cortes et al. (2004) is an effective way for motion cooperation. The CVT algorithm is a locational optimization approach based on a utility function. Here we propose to use the information entropy $H(\mathrm{x}_{*,t})$ as the utility function in CVT algorithm. The idea is to allow each node to move to the center of its Voronoi cell, which is weighted by the information entropy.

A Voronoi tessellation consists of multiple Voronoi cells $V_{i,t}$, each of which is occupied by a sensor nodes at time step t. A Voronoi cell $V_{i,t}$ is defined as follows:

$$V_{i,t} = \{\mathrm{x}_{*,t} \in Q \mid \|\mathrm{x}_{*,t} - \mathrm{x}_{i,t}\| \leq \|\mathrm{x}_{*,t} - \mathrm{x}_{j,t}\|, \forall i \neq j\}$$

CVT is a special Voronoi tessellation, which requires each sensor node move forward to the mass center of its Voronoi cell. The utility function of locational optimization problem is defined as:

$$U(\mathrm{x}_{1,t}, \ldots, \mathrm{x}_{N,t}) = \sum_{i=1}^{N}\int_{V_{i,t}}\frac{1}{2}\|\mathrm{x}_{*,t} - \mathrm{x}_{i,t}\|^2 H(\mathrm{x}_{*,t})d\mathrm{x}_{*,t}$$

For computing the mass center of each $V_{i,t}$, we use the following definitions:

$$M_{V_{i,t}} = \int_{V_{i,t}} H(\mathrm{x}_{*,t}) d\mathrm{x}_{*,t}$$
$$L_{V_{i,t}} = \int_{V_{i,t}} \mathrm{x}_{*,t} H(\mathrm{x}_{*,t}) d\mathrm{x}_{*,t}$$
$$C_{V_{i,t}} = \frac{L_{V_{i,t}}}{M_{V_{i,t}}}$$

The gradient of the utility function with respect to sensor node position $\mathrm{x}_{i,t}$ is:

$$\frac{\partial U}{\partial \mathrm{x}_{i,t}} = -\int_{V_{i,t}} (\mathrm{x}_{*,t} - \mathrm{x}_{i,t}) H(\mathrm{x}_{*,t}) d\mathrm{x}_{*,t}$$
$$= -M_{V_{i,t}} (C_{V_{i,t}} - \mathrm{x}_{i,t})$$

The control input signal of sensor node i is:

$$\mathrm{u}_{i,t} = k \frac{\partial U}{\partial \mathrm{x}_{i,t}}$$

where k is the control gain.

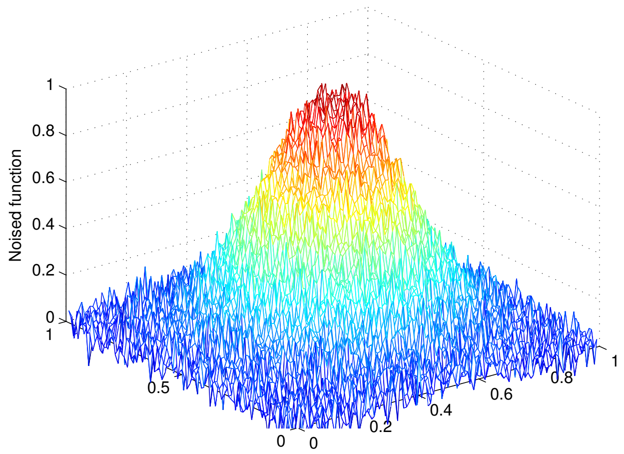
4. SIMULATIONS

An 1×1 area was used in simulations. The mobile sensor network consisted of $N = 30$ sensor nodes and they were randomly distributed in a small area with a size of 0.2×0.2. The hyperparameters learned are the lengthscale l_t and the amplitude a_t. The hyperparameter σ_t was given constant in simulations. A Gaussian like spatial function was simulated first. The sensor observation noise followed a Gaussian distribution with a standard deviation of $\sigma_t = 0.05$. The noised function is shown in Fig. 1(a). The predictive mean function is shown in Fig. 1(b). These two figures show that the proposed algorithm was able to model a simple spatial function.

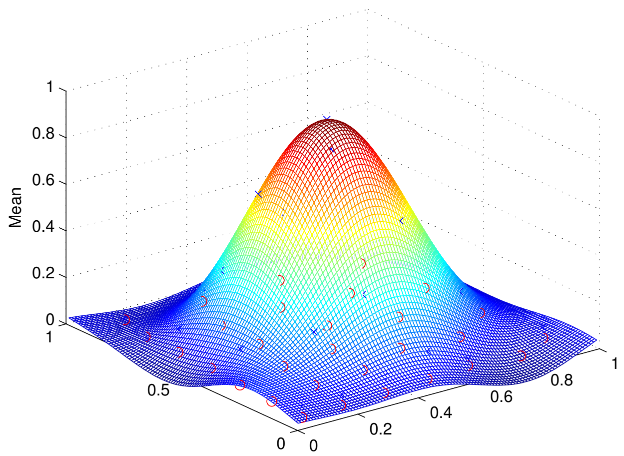
The mobile sensor trajectories are shown in Fig. 2(a). 30 sensor nodes were initially placed at the left bottom corner of the environment. They were able to move to cover as large as possible the area according to the uncertainty of the predictive model. The root mean squared error (RMSE) between the predictive function and the ground truth function with noise is shown in Fig. 2(b). At the beginning of the process (from loop 1 to loop 25), the RMSE was kept nearly unchanged. Then the RMSE experienced a sharp rise at around the 40th loop. It indicated an exploring behavior of the proposed algorithms given the noised observations. After the sharp rise, the RMSE declined rapidly and converged to zero. The behavior demonstrated in the changes of RMSE can also be observed from the results of hyperparameter learning. The lengthscale and the amplitude of squared exponential function were learned online using the maximum likelihood algorithm. The learned results are shown in Fig. 3(a) for lengthscale and in Fig. 3(b) for amplitude. Both parameters were kept very low due to the lack of enough coverage of the environment by all the sensor nodes at the beginning. Then a sharp rise was observed in both parameters. After the 40th loop, the hyperparameters declined to stable values. The stable value of lengthscale was about 0.25 and the stable value of amplitude was about 0.28.

A more complex $2D$ spatial function was simulated to show the proposed algorithm can also handle non-Gaussian spatial functions. The ground truth function is shown Fig. 4(a). The sensor noise was sampled with a standard deviation of $\sigma = 0.005$. The predictive mean function is shown in Fig. 4(b). By comparing the ground truth function with the predictive mean function, it is found that they were very similar. The large error was found at the boundary of the squared area. This is because the CVT algorithm kept the mobile sensor nodes inside of the Voronoi cell and the boundary information was less sampled.

The trajectories of mobile sensor nodes demonstrated the behavior of the proposed algorithm. They started from the corner of 0.2×0.2 and moved to the uncertainty region according to the information entropy. In the end, they covered the area in an optimized pattern shown in Fig. 5(a). The RMSE curve in Fig. 5(b) shows the convergent property of the whole process. All the sensor nodes did explore the environment at the beginning of the learning. Then the RMSE started to decline until a value very close to zero was reached.

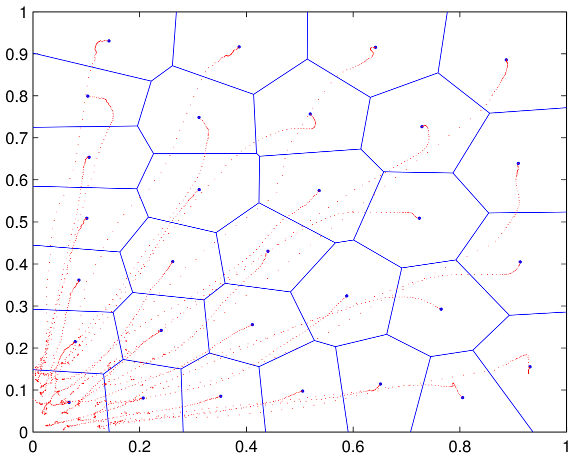


(a)

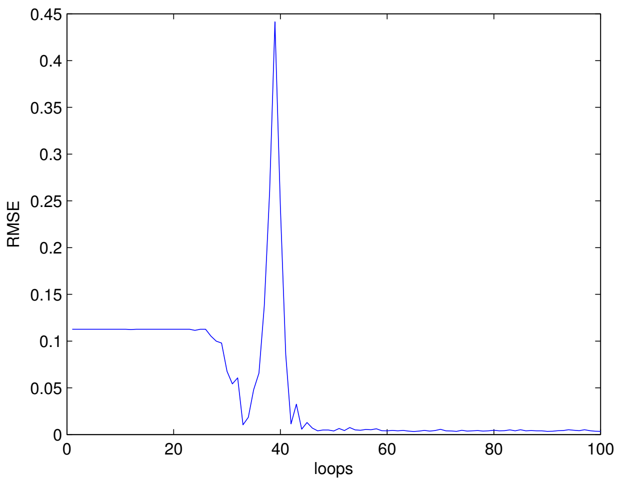


(b)

Fig. 1. Ground truth function with noise and predictive mean function

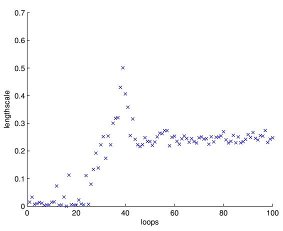


(a)

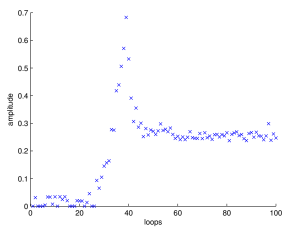


(b)

Fig. 2. Trajectories and RMSE changes



(a)

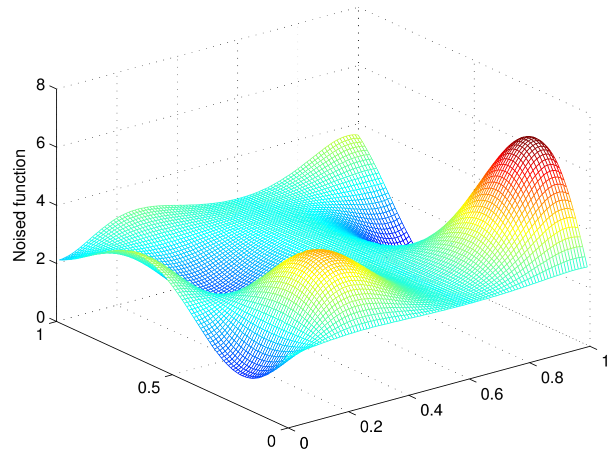


(b)

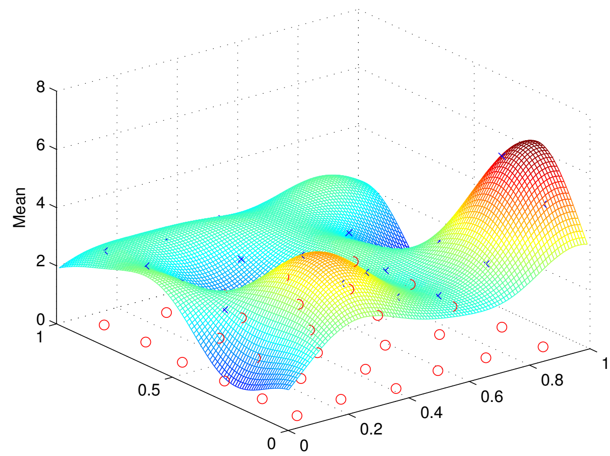
Fig. 3. Learning results of hyperparameters

5. CONCLUSIONS

GP regression is an effective tool for spatial function model problems due to its ability of generating predictive mean function and predictive covariance function for the function to be estimated. With the predictive covariance function, an information entropy based approach to motion decision making is proposed in this paper. Combining the information entropy with the CVT algorithm, it is able to explore the uncertainty of the environment and exploit the predictive model for multiple mobile sensor nodes. The proposed strategy is an active spatial function modeling approach.



(a)

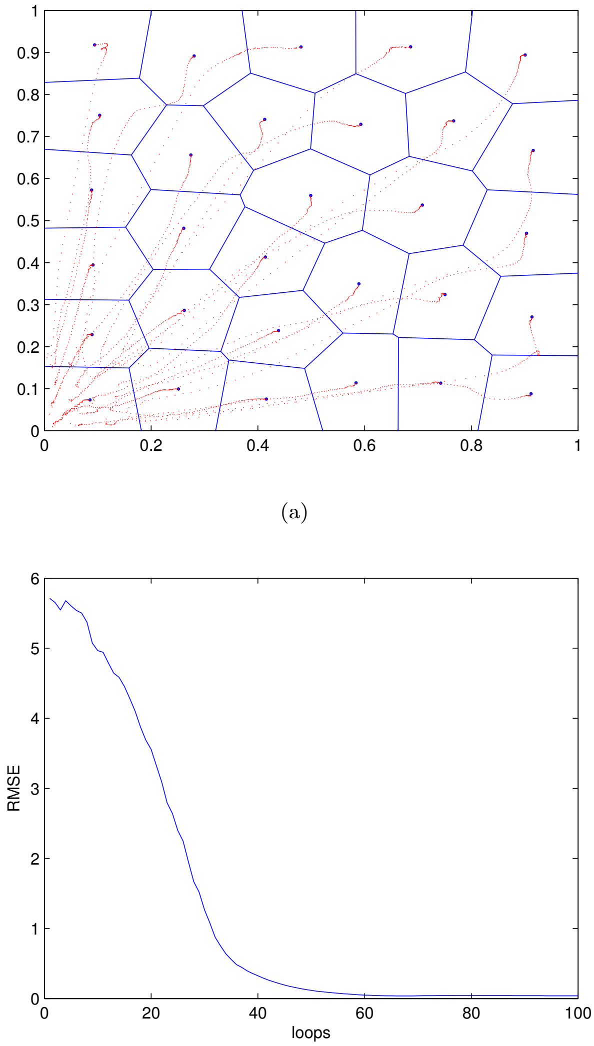


(b)

Fig. 4. Ground truth non-Gaussian function and predictive mean function

Since the information entropy is only related to the covariance function, not related to the sensor observation, it is possible to control mobile sensor nodes to move even without practical sensor observations. However, when the hyperparameters are learned, practical sensor observation is necessary for motion decision making. Although the proposed algorithm targets to spatial functions, not spatial-temporal functions, it is possible for the proposed approach to be applied to spatial-temporal functions where temporal dynamics can be handled implicitly by the adaptive covariance function learned from the online hyperparameter learning algorithm.

In the next step, we would like to test our proposed algorithm to spatial-temporal functions to check its performance. Alternatively, it is feasible to use a hierarchical Bayesian structure to explicitly handle temporal dynamics. In a hierarchical Bayesian structure, a low layer han-



(a)

(b)

Fig. 5. Trajectories and RMSE changes

dles the spatial dynamics using the proposed GP regression while a high layer handles the temporal dynamics using Kalman filter.

ACKNOWLEDGEMENTS

This research work is financially sponsored by European Union FP7 program, ICT-231646, SHOAL.

REFERENCES

Choi, J., Lee, J., and Oh, S. (2008). Swarm intelligence for achieving the global maximum using spatio-temporal Gaussian processes. In *Proc. of the American Control Conference*. Seattle, Washington.

Corrigan, C.E., Roberts, G.C., Ramana, M.V., Kim, D., and Ramanathan, V. (2007). Capturing vertical profiles of aerosols and black carbon over the indian ocean using autonomous unmanned aerial vehicles. *Atmospheric Chemistry and Physics Discussions*, 7(4), 11429–11463.

Cortes, J. (2009). Distributed Kriged Kalman filter for spatial estimation. *IEEE Trans. on Automatic Control*, 54(12), 2816–2827.

Cortes, J., Martinez, S., Karatas, T., and Bullo, F. (2004). Coverage control for mobile sensing networks. *IEEE Trans. on Robotics and Autonamous*, 20(2), 243–255.

Hussein, I. (2007). A Kalman filter-based control strategy for dynamic coverage control. In *Proc. of the 2007 American Control Conference*, 3271–3276. New York, NY.

Krause, A., Singh, A., and Guestrin, C. (2008). Near-optimal sensor placements in Gaussian processes: theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9, 235–284.

Leonard, N.E., Paley, D., Lekien, F., Sepulchre, R., Fratantoni, D.M., and Davis, R. (2007). Collective motion, sensor networks and ocean sampling. *Proceedings of the IEEE*, 95(1), 48–74.

Lynch, K.M., Schwartz, I.B., Yang, P., and Freeman, R.A. (2008). Decentralized environmental modelling by mobile sensor networks. *IEEE Trans. on Robotics*, 24(3), 710–724.

MacKay, D.J.C. (1998). Introduction to Gaussian processes. In C.M. Bishop (ed.), *Neural Networks and Machine Learning*, volume 168, 133–165. Springer, Berlin.

Martinez, S. (2010). Distributed interpolation schemes for field estimation by mobile sensor networks. *IEEE Trans. on Control Systems Technology*, 18(2), to appear.

Merino, L.F., Caballero, J.R., de Dios, J.M., and Ferruz, A.O. (2006). A cooperative perception system for multiple UAVs: application to automatic detection of forest fires. *Journal of Field Robotics*, 23(3), 165–184.

Ny, J.L. and Pappas, G. (2009). On trajectory optimization for active sensing in Gaussian process models. In *Proc. of the IEEE Conf. on Decision and Control*, 6286–6292. Shanghai, China.

Rasmussen, C.E. and Williams, C. (2006). *Gaussian Processes for Machine Learning*. MIT press, Cambridge MA.

Schwager, M., Rus, D., and Slotine, J. (2009). Decentralized, adaptive converage control for networked robots. *Int. J. of Robotics Research*, 28(3), 357–375.

Singh, A., Ramos, F., Whyte, H.D., and Kaiser, W.J. (2010). Modeling and decision making in spatio-temporal processes for environmental surveillance. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*. Anchorage, Alaska, USA.

Stachniss, C., Plagemann, C., and Lilienthal, A.J. (2009). Learning gas distribution models using sparse Gaussian process mixtures. *Autonomous Robots*, 26(2-3), 187–202.

Stranders, R., Rogers, A., and Jennings, N. (2008). A decentralized, on-line coordination mechanism for monitoring spatial phenomena with mobile sensors. In *Proc. of Second International Workshop on Agent Technology for Sensor Networks*. Estoril, Portugal.

Williams, C.K.I. and Rasmussen, C.E. (1996). Gaussian processes for regression. In D.S. Touretzky, M.C. Mozer, and M.E. Hasselmo (eds.), *Advances in Neural Information Processing Systems 8*, 514–520. MIT Press, Cambridge.