BACTERIAL FORAGING TECHNIQUES FOR SOLVING EKF-BASED SLAM PROBLEMS

Amitava Chatterjee¹,² and Fumitoshi Matsuno¹,³

¹Department of Mechanical Engineering And Intelligent Systems, The University of Electro-Communications, Chofu, Tokyo 182-8585, Japan
²Department of Electrical Engineering, Jadavpur University, Kolkata, West Bengal 700032, India
³International Rescue System Institute, Kawasaki, Kanagawa 210-0855, Japan

Abstract—The present paper proposes a successful application of bacterial foraging strategies to improve the quality of solutions that extended Kalman filters (EKFs) can offer to solve simultaneous localization and mapping (SLAM) problems for mobile robots and autonomous vehicles. The system can be especially useful in those situations where an incorrect knowledge of Q and R matrices of EKF can significantly degrade the SLAM performance. A fuzzy system has been implemented to adapt the R matrix of the EKF online, in order to improve its performance. The determination of the free parameters of the fuzzy system is configured as a high-dimensional metaheuristic optimization problem and E. coli bacterial swarm foraging has been successfully implemented to solve this optimization problem offline. The utility of the proposed system is aptly demonstrated by solving the SLAM problem for a mobile robot with varying landmarks and with wrong knowledge of sensor statistics. The proposed system could demonstrate enhanced performance in comparison with usual EKF-based systems for identical environment situations. Copyright © 2002 USTARTH

Keywords: extended Kalman filter, fuzzy based adaptation, sensor statistics, SLAM problem

I. INTRODUCTION

The simultaneous localization and mapping (SLAM) problem has been regarded as a complex and important present day research problem within the research community of mobile robots and autonomous vehicles. The problem under consideration is the navigation of a mobile platform in an environment where both the map of the environment and the localization of the mobile platform are unknown. The problem is solved iteratively where both localization of the robot/vehicle and determination of the environment map are estimated simultaneously [1], [2]. Some successful solutions for this complicated problem have so far been proposed employing more traditional extended Kalman filter [3-6] and more recent particle filter [2] based algorithms. In EKF based solution, the problem is solved by maintaining a complete state vector of the robot pose and the position of each landmark observed in the environment. The uncertainties associated with this estimation process are kept stored in covariance matrices. The solution of such a problem by configuring it as a state estimation problem is more complicated than conventional state estimation problems because the state vector is dynamic in size and the number of landmarks observed and which landmarks are observed varies with time instants. Furthermore, there is always the possibility that both the vehicle’s pose estimate and its associated map estimates become increasingly inaccurate in absence of any global position information [2].

The performance of an EK-based SLAM problem will depend on how accurately the statistics depicting measurement uncertainties and process uncertainties are known. EKF assumes independent Gaussian distribution based measurement noise and process noise models with known covariance and these are specified in form of process covariance matrix (Q) and measurement noise covariance matrix (R). An incorrect a priori knowledge of Q and R (which is quite possible in real world scenario) may lead to performance degradation [7] and it can even lead to practical divergence [8]. In fact, for SLAM problems our experiments have shown that the performance can become significantly worse under these situations. In our scheme, we propose a fuzzy logic based system, which will adapt the R matrix of EKF-based SLAM problem online, so that the solution offered by the EKF can be kept satisfactory.

Although fuzzy logic has been employed previously to tune traditional Kalman filters with fixed number of states by adapting both Q and R simultaneously or adapting R only [9,10,12], our algorithm considers a much more complicated problem in SLAM domain where the sizes of the state vector and hence the covariance matrix keep varying with iterations. Hence it is very important that the free parameters of the fuzzy system are tuned properly so

Amitava Chatterjee’s work was supported by FY 2004 JSPS Post-doctoral Fellowship for Foreign Researchers in Japan. He was on leave from Jadavpur University, Kolkata, West Bengal 700032, India.

Amitava Chatterjee is with Jadavpur University, Kolkata, West Bengal 700032, India. He was with The University of Electro-Communications, Chofu, Tokyo 182-8585, Japan when this work was performed (e-mail: achatterjee @ee.jdvu.ac.in).

Fumitoshi Matsuno is with The University of Electro-Communications, Chofu, Tokyo 182-8585, Japan and International Rescue System Institute, Kawasaki, Kanagawa 210-0855, Japan (e-mail: matsuno@hi.mce.uec.ac.jp).
that the fuzzy adaptation, online, can provide meaningful supervision of the performance of EKF for SLAM problems. Hence we employed a recently proposed stochastic global optimization technique, called bacterial foraging strategy, to learn the fuzzy system parameters. This optimization strategy is based on the concept that for those animals which can locate, handle and ingest food better than other animals, the propagation of genes is favored and they are more likely to enter into a reproduction mechanism [11]. The chemotactic behavior of E. coli bacteria has been biomimicked to successfully evolve an optimization strategy [11], [12]. These types of stochastic methods are especially suitable for the problem that we have considered, because it is a non-gradient based optimization technique and it is very difficult to employ gradient-based techniques for learning the fuzzy system that we have under our consideration.

The rest of the paper is organized as follows. Section II presents the fuzzy adapted EKF-based stochastic SLAM algorithm. Section III presents a short primer on the bacterial foraging strategy that have been successfully employed to learn the fuzzy system. Section IV presents the stochastic methods are especially suitable for the problem that we have considered, because it is a non-gradient based optimization technique and it is very difficult to employ gradient-based techniques for learning the fuzzy system that we have under our consideration.

The rest of the paper is organized as follows. Section II presents the fuzzy adapted EKF-based stochastic SLAM algorithm. Section III presents a short primer on the bacterial foraging strategy that have been successfully employed to learn the fuzzy system. Section IV presents the stochastic methods are especially suitable for the problem that we have considered, because it is a non-gradient based optimization technique and it is very difficult to employ gradient-based techniques for learning the fuzzy system that we have under our consideration.

The rest of the paper is organized as follows. Section II presents the fuzzy adapted EKF-based stochastic SLAM algorithm. Section III presents a short primer on the bacterial foraging strategy that have been successfully employed to learn the fuzzy system. Section IV presents the stochastic optimization technique and it is very difficult to employ gradient-based techniques for learning the fuzzy system that we have under our consideration.

ii) Prediction Step
In each prediction step, the augmented state vector estimate \( \hat{x}_a \) in the next sampling instant is predicted on the basis of the augmented state vector at the present sampling instant \( \hat{x}_a \) and the change in vehicle pose between successive sampling instants (separated by the discrete time step \( \Delta t \)). The control input vector \( u = [v, s]^T \) is composed of the steering angle command (s) and the velocity at which the rear wheel of the robot is driven (v). So the state estimates can be obtained by employing wheel encoder odometry and the robot kinematic model. However, the control inputs v and s must be considered with their uncertainties involved due to e.g. wheel slippage, incorrect calibration of vehicle controller etc. These uncertainties are modeled as Gaussian variations in v and s from their nominal values. Hence, the prediction step calculates: 
\[
\hat{x}_a = f(\hat{x}_a, u) + \mathbf{v}_a,
\]
where the control inputs u are given by: 
\[
\mathbf{v}_a = \begin{bmatrix} v \\ s \end{bmatrix}
\]
and the change in vehicle pose between successive sampling instants (separated by the discrete time step \( \Delta t \)).

The Jacobians and U, the covariance matrix of u, are given as:
\[
\mathbf{V}_x = \frac{\partial f}{\partial \mathbf{x}} \bigg|_{(\hat{x}_a, u)}, \mathbf{V}_u = \frac{\partial f}{\partial \mathbf{u}} \bigg|_{(\hat{x}_a, u)}, 
\]
where the Jacobians and U, the covariance matrix of u, are given as:
\[
\mathbf{U} = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_s^2 \end{bmatrix}
\]
Because the features are assumed to remain stationary with time, \( \hat{x}_m \) and \( \mathbf{P}_m \) in the previous equations remain constant with time.

ii) Update Step
Let us assume that we re-observe an existing feature i.e. the \( i \)th feature \( (\hat{x}_i, \hat{y}_i) \). Let this feature be measured in terms of its range (r) and bearing (\( \theta \)) relative to the observer, given as:
\[
\mathbf{z} = [r, \theta]^T
\]
The uncertainties in these observations are again modeled by Gaussian variations and let \( \mathbf{R} \) be the corresponding observation/measurement noise covariance matrix given as:
\[
\mathbf{R} = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix}
\]
where we assume that there is no cross-correlation between the range and bearing measurements. In the context of the map, the measurements can be given as:

$$\tilde{z}_t = h_t(\tilde{x}_t, y_t) = \left[ \sqrt{(\tilde{x}_t - x_t)^2 + (\tilde{y}_t - y_t)^2} \right] \arctan\left( \frac{\tilde{y}_t - y_t}{\tilde{x}_t - x_t} \right)$$

(13)

Now the Kalman gain $W_i$ can be calculated assuming that there is no cross-correlation between $z$ and $(\tilde{x}_t, \tilde{y}_t)$ and the following computations can be resorted to:

$$V_i = z - h_t(\tilde{x}_t)$$

(14)

$$S_i = \nabla h_x \sigma_a^T + R$$

(15)

$$W_i = P_a^{-1} \nabla h_x^T S_i^{-1}$$

(16)

where $V_i$ denotes the innovation of the observation for this $i$th landmark and $S_i$ the associated innovation covariance matrix. The Jacobian $\nabla h_x$ is given as:

$$\nabla h_x = \frac{\partial h_t}{\partial x_a}$$

(17)

Hence, the a posteriori augmented state estimate and the corresponding covariance matrix are updated as:

$$\hat{x}_a^+ = \hat{x}_a^+ + W_i V_i$$

(18)

$$P_a^+ = P_a^- - W_i S_i W_i^T$$

(19)

However, during the execution of SLAM algorithm, it is highly likely that a new feature is observed for the first time. Then this should be initialized into the system by incorporating their 2-D position coordinates in the augmented state vector and accordingly modifying the covariance matrix. These new $\hat{x}_a^+$ and $P_a^+$ can be calculated as:

$$\hat{x}_a^+ = \left[ \begin{array}{c} \hat{x}_a^- \\ f(\tilde{x}_a, z) \end{array} \right]$$

(20)

$$P_a^+ = \left[ \begin{array}{cc} P_a^- & \nabla g^T \sigma_a^T \\ \nabla g & \sigma_a^T \end{array} \right]$$

(21)

Here $f(\tilde{x}_a, z)$ is employed to convert the polar observation $z$ to the base Cartesian coordinate frame. The Jacobians are calculated as:

$$\nabla g = \frac{\partial f}{\partial x_a}(\tilde{x}_a, z), \quad \nabla g_z = \frac{\partial f}{\partial z}(\tilde{x}_a, z)$$

(22)

Similarly, we can delete any unreliable feature by deleting the relevant row entry from the state vector and the relevant row and column entries from the covariance matrix.

Now, a fuzzy adaptation scheme is proposed to adopt the R matrix, online, to improve the performance of the EKF. The concept utilized is similar to the fuzzy logic based innovation adaptive estimation (IAE) approach [10], where new statistical information from innovation sequence is utilized to correct the estimation of the states. The basic concept relies on determining the discrepancy between a new measurement $z_k$ and its corresponding predicted estimation $\hat{z}_k$, at any arbitrary $k$th instant, and utilizing this new information to correct the estimations/predictions already made. In our SLAM algorithm, actual covariance of the innovation sequences, $C_{\text{innk}}$, is calculated as:

$$C_{\text{innk}} = \nu_k \nu_k^T$$

(23)

where $\nu_k$ denotes the augmented innovation sequence, made consistent with the batch mode of observations. The objective is to minimize the mismatch between $C_{\text{innk}}$ and the theoretical covariance of the innovation sequences $(S_k)$, made consistent with the batch mode of observations. The mismatch, at the $k$th instant, is given as:

$$\Delta C_{\text{innk}} = C_{\text{innk}} - S_k$$

(24)

Compared to conventional EKF algorithms, the EKF-based SLAM is much more complex, because size of $\nu_k$ and $C_{\text{innk}}$ keep changing from iteration to iteration because each of them is dependent on the number of features observed during any given observation and update step, which were all observed at least once before.

Now, a fuzzy system is employed to minimize the mismatch given in (24). The overall fuzzy system employs a bank of subsystems where each subsystem employs an SISO fuzzy system. These SISO fuzzy systems are utilized to use each diagonal element of the $\Delta C_{\text{innk}}$ matrix, i.e. $\Delta C_{\text{innk}}(j, j)$, as the input and the output is $\Delta R(j, j)$, an adaptation recommended for the corresponding diagonal element of the augmented measurement noise covariance matrix $R$ matrix, computed according to the batch-mode situation. The size of the augmented $R$ matrix varies with iteration and it is $[2z_f \times 2z_f]$ where $z_f$ is the number of landmarks observed in that iteration, which were also observed earlier. This augmented $R$ is formed utilizing the original $[2 \times 2]$ $R$ matrix and this is formulated as:

$$\text{augmented } R = \left[ \begin{array}{cccc} \sigma_{\rho r}^2 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \sigma_{\theta r}^2 & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \ddots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \sigma_{\rho r}^2 & 0 \\ 0 & 0 & \cdots & 0 & \sigma_{\theta r}^2 \\ \end{array} \right]$$

(25)

Here, $\sigma_{\rho r}^2$ and $\sigma_{\theta r}^2$ correspond to the sensor statistics computed for that iteration. We employed the same fuzzy system for each of those SISO fuzzy subsystems to simplify the design and hence we employed normalized input for each fuzzy subsystem. However, the ranges of mismatches in range and bearing observations may be of different order depending on the degree of accuracy/inaccuracy in the available knowledge of those specific sensor statistics. Hence we segregated $\Delta C_{\text{innk}}(j, j)$ elements corresponding to range and bearing observations in two different sets and employed normalization of each fuzzy system input corresponding to range and bearing observations separately. The input to each fuzzy subsystem is fuzzified employing three membership functions (MFs):
negative (N), zero (Z) and positive (P), as shown in fig. 1. Then each system employs three fuzzy rules:

\[
\begin{align*}
\text{IF } A_{\text{C,nabr}}(f_j) \text{ is } N & \quad \text{THEN } \Delta R(j, j) = w_1, \\
\text{IF } A_{\text{C,nabr}}(f_j) \text{ is } Z & \quad \text{THEN } \Delta R(j, j) = w_2, \\
\text{IF } A_{\text{C,nabr}}(f_j) \text{ is } P & \quad \text{THEN } \Delta R(j, j) = w_3.
\end{align*}
\]

The final output of each fuzzy system i.e. \( \Delta R(j, j) \) is calculated by employing well known weighted average technique. Finally the adaptations, i.e. \( \Delta \sigma_\theta^2 \) and \( \Delta \sigma_\theta^2 \), required for the original \([2\times2]\) \( R \) matrix are computed on the basis of appropriate means, separately computed from the arrays of \( \Delta R(j, j) \) entries for range and bearing observations. This adapted original \([2\times2]\) \( R \) matrix is employed to prepare the augmented \( R \) in (25) and it is utilized in the next observation and update step for the EKF-based SLAM algorithm.

![Membership functions chosen for each fuzzy subsystem.](image)

Now, for satisfactory performance of the fuzzy adapted EKF, it is very important that the free parameters of the fuzzy system are properly chosen. These free parameters include the parameters of the three MFs and the three output gains of the three fuzzy rules for each SISO fuzzy subsystem. One of the prime difficulties encountered in training these parameters for the fuzzy systems is that the desired output for a given input for each of these fuzzy systems is not definitely known and hence we can not employ usual supervised mode of neural network training to determine these parameters. Hence we employed bacterial foraging strategy, a new technique in the category of evolutionary methods for stochastic global optimization problems. These methods can be very suitable in those situations where the optimization technique can be employed for minimization of a function without the knowledge of the goal i.e. the minimum value it is supposed to attain finally and they do not require any gradient information for their operation. Here, the free parameters of the fuzzy system are organized in form a vector \( x = (N_i, N_b, Z_i, Z_b, P_i, P_b) \) and we attempt to minimize a cost function \( f(x) \) on the basis of the \( x \). The cost function is designed as:

\[
f = \frac{\sum_{j=1}^{J_{C,nabr}} (A_{C,nabr}(f_j))^2}{\sum_{j=1}^{J_{C,nabr}} J_{C,nabr}}
\]

where \( N_{obs} \) denotes the total number of observation instants in a given iteration and \( J_{C,nabr} \) denotes the total number of diagonal elements of \( A_{C,nabr} \) matrix when the \( n_{obs} \)th observation is made. While implementing this optimization strategy, the variables forming the high-dimensional vector \( x \) are implemented with several constraints to satisfy some shape constraints chosen for the MFs and also to ensure that there is some overlapping between the stretches of consecutive MFs. These constraints are given as:

\[
\begin{align*}
\{ N_b < N_i \} & \quad \text{THEN } N_b = N_i, \\
\{ Z_i < Z_b \} & \quad \text{THEN } Z_i = Z_b, \\
\{ Z_l < Z_r \} & \quad \text{THEN } Z_l = Z_r, \\
\{ P_l < P_b \} & \quad \text{THEN } P_l = P_b, \\
\{ N_b < Z_b \} & \quad \text{THEN } N_b = Z_b, \\
\{ Z_b < P_b \} & \quad \text{THEN } Z_b = P_b.
\end{align*}
\]

### III. E. coli BACTERIAL FORAGING OPTIMIZATION

In foraging theory, it is assumed that the objective of the animals is to search for and obtain nutrients in such a fashion that the energy intake per unit time is maximized [11]. This foraging problem has been formulated as an optimization problem by employing optimal foraging theory. The foraging behavior of \( E. \ coli \) bacteria present in our intestines, which includes the methods of locating, handling and ingesting food, has been successfully mimicked to propose a new evolutionary optimization algorithm [11]. This optimization procedure comprises of four basic steps: a) chemotaxis, b) swarming, c) reproduction and d) elimination and dispersal. The optimization technique consists of determining the minimum of a function \( J(\theta) \) where the variables under consideration constitute the high-dimensional vector \( \theta \in \mathbb{R}^p \) and it is very difficult or almost impossible to determine \( \nabla J(\theta) \). Here \( \theta \) determines the position of a bacterium in high dimensional space. A negative value of \( J(\theta) \) indicates that the bacterium is in nutrient-rich environment, a zero value indicates a neutral environment and a positive value indicates a noxious environment. The objective will be to try and implement a biased random walk for each bacterium where it will try to climb up the nutrient concentration and try and avoid noxious substances and will attempt to leave a neutral environment as soon as possible.

In chemotaxis step, each bacterium either experience a tumble followed by another tumble or a tumble followed by a run. Let \( \theta(j,k,l) \in \mathbb{R}^p \) represent the \( i \)th bacterium in \( i = 1, 2, ... S \) in the \( j \)th chemotactic, \( k \)th reproduction and \( l \)th elimination-dispersal step. Let \( J(i,j,k,l) \) represent the cost associated with this position of the bacterium, although, strictly speaking, in terms of the notations used in the domain of foraging theory, \( J \) is known as the nutrient surface. The new position of a bacterium in the chemotaxis step is given as,

\[
\theta(j+1,k,l) = \theta(j,k,l) + C(i)\phi(j)
\]

where \( \phi(j) \) denotes a unit length random direction to represent the tumble and determine the future direction of movement and \( C(i) \) denotes the run-length i.e. chemotactic step size. If the cost \( J \) at \( \theta(j+1,k,l) \) gets lower and lower than the cost at just the preceding position i.e. \( \theta(j,k,l) \), then the bacterium will keep taking successive steps in that direction (constrained only by the maximum number of permissible successive steps \( N_i \)).

In swarming step, we model the attraction-repulsion behavior of a group of \( E. \ coli \) cells swimming together. Here cell-released attractants are employed by each cell to
signal other nearby cells to swarm with it. At the same time, the same cell can repel a nearby cell by consuming nearby nutrients and two cells can not simultaneously physically be present at the same location. This combined cell-to-cell attraction and repelling effect is given by

$$\sum_{j=1}^{S} J_{cc}(\theta_{j}, \theta_{k})$$

$$= \sum_{j=1}^{S} \left[ -d_{\text{attract}} \exp(-w_{\text{attract}} \sum_{m}^{\rho}(\theta_{m} - \theta_{m}^{j})^{2}) \right]$$

$$+ \sum_{j=1}^{S} h_{\text{repellant}} \exp(-w_{\text{repellant}} \sum_{m}^{\rho}(\theta_{m} - \theta_{m}^{j})^{2}) \right] \quad (29)$$

which is added to the actual cost function $J$ to be minimized and this presents a time-varying nature for $J$. The $d_{\text{attract}}$, $w_{\text{attract}}$, $h_{\text{repellant}}$, and $w_{\text{repellant}}$ parameters will determine the shape of these attraction and repulsion signals.

In reproduction step, the bacteria population is sorted in ascending order of accumulated cost and the 50% of least healthy bacteria die and each of the remaining 50% healthier bacteria split into two bacteria, such that each two child bacteria thus reproduced are placed at the same location. In this process those bacteria, which could search for a lot of nutrients, are rewarded and simultaneously, $S$, the bacteria population is kept constant.

In elimination-dispersal step, each bacterium is subjected to elimination-dispersal with a probability of $p_{\text{dis}}$, where the bacterium may be dispersed into an unexplored region of environment or search space. While this may destroy the progress achieved through the chemotactic process thus far, it may happen that the bacterium may find itself closer to new source of nutrients. After every $N_{c}$ chemotactic steps are completed, one reproduction step is undertaken and after $N_{r}$ reproduction steps are completed, one elimination-dispersal step is undertaken. Hence a bacterium will undergo several chemotactic steps before it is allowed to reproduce and the system will undergo several generations before an elimination-dispersal step is executed.

V. PERFORMANCE EVALUATION

The proposed solution for the SLAM problems has been tested for the benchmark environments, with varied number and positions of the landmarks, available in [13]. The package developed in [13] for conventional EKF-based SLAM solutions, employing static $Q$ and $R$ matrices, show that the system gives satisfactory solution for the sensor statistics, specified as $\sigma_{r} = 0.1$ m. and $\sigma_{h} = 1$ deg. The environment has been tested with varied degree of complexity comprising of 35, 135 and 497 landmarks respectively and in each case the robot is asked to navigate through 17 specified waypoints.

However, the performance of the same system degrades considerably when we consider that these sensor statistics are wrongly known. Let us assume that these statistics are known as: $\sigma_{r} = 0.01$ m. and $\sigma_{h} = 3.0$ deg. Then the performances exhibited by the conventional EKF-based SLAM [13] are shown in fig. 2(a) to fig. 2(c). It can be seen that these performances degraded significantly, specially when there are smaller number of landmarks in the environment. It can be seen that, when we employed our proposed system, the fuzzy adaptation could improve the performance satisfactorily in each case (shown in fig. 3(a)-3(c)). In each figure, the green lines depict the actual path traversed by the robot, while the black lines depict the SLAM estimated path traversed based on estimated states of robot poses in each sampling instant or iteration. The stars (+), in blue, depict the actual stationary landmark positions. The crosses (+), in red, depict the positions of these landmarks estimated at the end of the test run. It can be seen that for the fuzzy adapted algorithm, the estimated and actual robot paths are very close to each other and also most of the estimated landmarks very closely matched with the actual positions of the landmarks.

VI. CONCLUSION

The present paper proposed the development of a fuzzy adapted EKF-based SLAM algorithm, which can improve the performance under those situations where the sensor statistics are incorrectly known a priori. The system can improve the performance by adapting the $R$ matrix online. For useful performance of the fuzzy adaptation we have successfully employed a bacterial foraging strategy, offline, to train its parameters. Several experiments in simulation could confirm the usefulness of the proposed scheme.

REFERENCES

Fig. 2. Performance of the conventional EKF-based SLAM \((\sigma_x = 0.01 \text{ m. and } \sigma_\delta = 3.0 \text{ deg.})\) with (a) 35, (b) 135 and (c) 497 features/landmarks in the environment.

Fig. 3. Performance of the fuzzy adapted EKF-based SLAM \((\sigma_x = 0.01 \text{ m. and } \sigma_\delta = 3.0 \text{ deg.})\) with (a) 35, (b) 135 and (c) 497 features/landmarks in the environment.