Online ISSN 1848-3380, Print ISSN 0005-1144
ATKAFF 56(4), 466–477(2015)

Peiliang Wu, Lingfu Kong, Liang Kong, Shihui Zhang

Research on Simultaneous Localization, Calibration and Mapping of Network Robot System

DOI 10.7305/automatika.2016.01.710
UDK 004.896:528.023; 519.245/.246

Original scientific paper

In a network robot system, a robot and a sensor network are integrated smoothly to develop their advantages and benefit from each other. Robot localization, sensor network calibration and environment mapping are three coupled issues to be solved once network robot system is introduced into a service environment. In this article, the problem of simultaneous localization, calibration and mapping is raised in order to improve their precision. The coupled relations among localization, calibration and mapping are denoted as a joint conditional distribution and then decomposed into three separate analytic terms according to Bayesian and Markov properties. The framework of Rao-Blackwellized particle filtering is used to solve the three analytic terms, in which extended particle filter is used for localization and unscented Kalman filter is used for both calibration and mapping. Simulations have been performed to demonstrate the validity and efficiency of the proposed solutions.

Key words: network robot system, simultaneous localization, calibration and mapping, Rao-Blackwellized particle filtering, unscented transformation

Istraživanje simultane lokalizacije, kalibracije i kartiranja umreženim robotskim sustavima. U umreženom robotskom sustavu, robot i senzorska mreža su međusobno integrirani i povezani na način da i jedan i drugi iskoriste svoje prednosti, te da imaju koristi jedan od drugoga. Kako bi umreženi robotski sustav mogao djelovati u radnom okruženju potrebno je riješiti tri međusobno povezana problema: lokalizaciju, kalibraciju senzorske mreže i kartiranje prostora. U ovom radu razmatraju se problemi istodobne lokalizacije, kalibracije i kartiranja te se razmatraju mogućnosti poboljšanja njihove preciznosti. Povezanost lokalizacije, kartiranja i kalibracije predstavljena je pomoću zajedničke uvjetne razdiobe i zatim rastavljena u tri razdvojena analitička izraza korištenjem Bayesovih i Markovljevih svojstava. Za rješavanje svih triju analitičkih izraza koristi se Rao-Blackwell čestično filtriranje, pri čemu se prošireni čestični filtar koristi kod lokalizacije a nederivirajući Kalmanov filtar za kalibraciju i kartiranje. Ispravnost i efikasnost predloženog pristupa pokazana je kroz provedene simulacije.

Ključne riječi: umreženi robotski sustav, istovremena lokalizacija, kalibracija i kartiranje, Rao-Blackwellized čestični filtar, transformacija

1 INTRODUCTION

Network robot system, which is also called as ubiquitous robot system in some articles [1-3], is a new robotic form in which a robot and a sensor network are integrated smoothly to develop their advantages and benefit from each other [4-5]. On one hand, the sensor network extends the effective sensing range of the robots, on the other hand, the robots in turn can calibrate, deploy and maintain the sensor network to increase its longevity and utility. Hence, the network robot system has important application perspective in large-area and especially dynamic service environment, for example, home monitoring [6], distributed air pollution sensing [7], and other urban environment [8].

Once a network robot system is introduced into a service environment, three issues will be solved in order to make it work, which are the robot localization, sensor network calibration and environment mapping. Furthermore, three issues are coupled with each other as shown in Fig. 1, that is, (i) the robot can be localized according to the robot control, calibrated sensor network observations robot and robot observations mapped environment features, (ii) the sensor network can be calibrated according to its observations both robot path and mapped environment features, (iii) the environment can be mapped according to both robot observations and calibrated sensor network observations. The solution of each issue can be optimized by referring to other two, especially in a view of that both control errors and observation errors exist anywhere and anytime in reality. When only solving the coupled relation between robot localization and environment mapping,

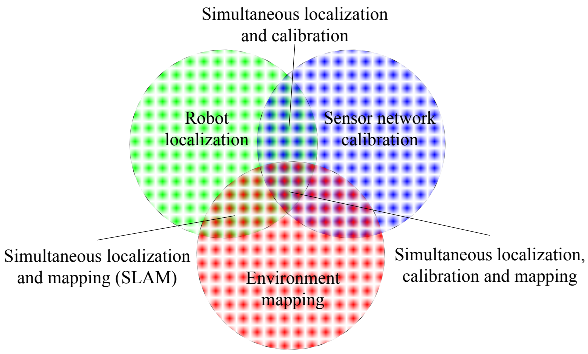

Fig. 1. Coupled relations between robot localization, sensor network calibration and environment mapping

it degenerates into the well-known simultaneous localization and mapping (SLAM) problem [9].

Recently researches pay more attention to simultaneous localization and calibration problem. Wu provided a method of simultaneous robot localization and camera-network calibration using Rao-Blackwellized Particle Filter [3]. Chen put forward an approach to calibrate camera-network online base on its observations of robot path [10], and then Liang came up with an approach of Monte Carlo-based mobile robot localization by distributed perceptive cooperation, which assists robot localization by considering the observations from the calibrated sensor network [11]. Caballero presented an approach of wireless sensor network calibration based on mobile robot [12]. Herrero focused on indoor localization of a mobile robot in a WSN using only inter-node range measurements, and proposed a range-only fuzzy Voronoi-enhanced localization method [13].

Environment mapping with network robot system is mainly researched under the SLAM framework in existent literature. Tuna presented the communication related design considerations of wireless sensor network aided multi-robot SLAM [14]. Schleicher presented a real-time hierarchical SLAM system based on stereovision and GPS fusion [15]. Menegatti presented the localization of a mobile robot while simultaneously mapping the location of the nodes of a wireless sensor network using only range measurements [16]. Wong looked into the problem of data association for SLAM in the robotic wireless sensor network environment [17]. Angel addressed the problem of simultaneously localizing multiple targets and estimating the positions of the sensors in a sensor network using particle filters [18], and Liang gave a distributed- perception-based robot SLAM approach to build an accurate map of large-scale environment [19]. Park gave a map-building and localization by three-dimensional local features for ubiquitous service robot [20]. Kim presented an ubiquitous SLAM method for robots with ambient intelligence [21]. Although all the above methods have given solutions to the sensor network-assisted robot SLAM, the calibration of the sensor network, which is the foundation of the sensor network assisted robot SLAM, has not yet been considered simultaneously. Kelly combined visual and inertial sensors into a robot, and presented a visual-inertial simultaneous localization, mapping and robot onboard sensor self-calibration [22], and some multi-robot distributed SLAM algorithms have been proposed [23,24], but in those papers sensors are installed on these robots and move with them, so the error of sensor observations will determinately increase with robot motions.

In this article, considering the coupled relations among robot localization, sensor network calibration and environment mapping, the concept of simultaneous localization, calibration and mapping of the network robot system is raised, which is denoted as a joint conditional distribution and then decomposed into several analytic terms in order to be solved separately under the framework of Rao-Blackwellized particle filtering.

2 SYSTEM STATEMENT

2.1 System construction and simplification

The network robot system discussed in this article consists of three parts: (1) sensor network with processing unit, (2) mobile service robot interacting with sensor network processing unit wirelessly, and (3) various targets in environment (including family persons, manipulated objects and environment landmarks and so on, all of them are referred as features throughout this article).

As a very important place of people's daily life, the layout structure of home or office is always relatively complicated. To analyze conveniently, some simplification is proposed in this article as follows. We take the robot coordinates at the beginning of mapping as world coordinates, and assume that robot moves in the plane parallel to the ground. In addition, environmental features are considered as points distributed in the same plane. So the whole environment will be described as a feature map under the world coordinates. Without loss of generality, we assume that the robot observes exactly one landmark at a time, and each sensor node also observes exactly one landmark around it.

2.2 System modeling

Firstly, a robot motion is modeled. Suppose that a robot moves in the x-y plane of the world coordinate system with pose $\boldsymbol{s}_t = \{x_t, y_t, \varphi_t\}$ at t-th moment, where x_t, y_t are x and y coordinate of the robot location respectively and φ_t is the robot orientation. Taking motion error into account, the motion model can be expressed in probability form as follows:

$$p\left(\boldsymbol{s}_t|\boldsymbol{s}_{t-1}, \boldsymbol{u}_t\right) = h\left(\boldsymbol{s}_{t-1}, \boldsymbol{u}_t\right) + \varepsilon_{h,t} \tag{1}$$

where $\boldsymbol{u}_t$ is the control at *t*-th moment, $h(\boldsymbol{s}_{t-1}, \boldsymbol{u}_t)$ is the ideal motion equation, $\varepsilon_{h,t}$ is white Gaussian noise submitting to distribution $N(0, \boldsymbol{P}_t)$.

Secondly, a robot observation is modelled. Suppose $\boldsymbol{\theta}_n$ as the *n*-th environment feature, and the robot observation to $\boldsymbol{\theta}_n$ is $\boldsymbol{y}_t^n$ at *t*-th moment, so the robot observation model of the environment features can be expressed as:

$$p(\boldsymbol{y}_t^n|\boldsymbol{\theta}_n, \boldsymbol{s}_t) = g(\boldsymbol{\theta}_n, \boldsymbol{s}_t) + \varepsilon_{g,t} \quad (2)$$

where $g(\boldsymbol{\theta}_n, \boldsymbol{s}_t)$ is the ideal equation of robot observation to *n*-th environment feature $\boldsymbol{\theta}_n$, $\varepsilon_{g,t}$ is white Gaussian noise submitting to $N(0, \boldsymbol{R}_t)$. From initial to *t*th moment, robot observations of the *N* environment features can be expressed as $\boldsymbol{y}^{t,N} = \{\boldsymbol{y}_i^n\}_{i=1,\cdots,t;n=1,\cdots,N}$.

Thirdly, a sensor network nodes observation of the environment feature is modelled. Suppose that the sensor network is composed of *M* sensor nodes, the location parameters of the *m*-th node are regarded as vector $\boldsymbol{\psi}_m$ in the world coordinate system, and its observation to $\boldsymbol{\theta}_n$ is regarded as $\boldsymbol{x}_n^m$. Taking sensor observation errors into account, the sensor $\boldsymbol{\psi}_m$ observation model of the environment feature $\boldsymbol{\theta}_n$ can be expressed as:

$$p(\boldsymbol{x}_n^m|\boldsymbol{\theta}_n, \boldsymbol{\psi}_m) = d(\boldsymbol{\theta}_n, \boldsymbol{\psi}_m) + \varepsilon_{d,t} \quad (3)$$

where $d(\boldsymbol{\theta}_n, \boldsymbol{\psi}_m)$ is the ideal equation of sensor ψ_m observation to environment feature $\boldsymbol{\theta}_n$, $\varepsilon_{d,t}$ is white Gaussian noise submitting to $N(0, \boldsymbol{T}_t)$. From initial to *t*-th moment, the sensor network observations to environment feature can be expressed as $\boldsymbol{x}^{t,M,N} = \{\boldsymbol{x}_n^m\}_{m=1,\cdots,M;n=1,\cdots,N}$.

Finally, the sensor network nodes observation to robot is modelled. As mentioned above, the pose of robot consists of location and orientation, both of which could be characterized by some identifiable tags [3]. Without loss of generality, we denote $\boldsymbol{z}_t^m$ as observation from sensor $\boldsymbol{\psi}_m$ to robot pose $\boldsymbol{s}_t$. Taking sensor observation errors into account, the observation model from sensor ψ_m to robot can be expressed as:

$$p(\boldsymbol{z}_t^m|\boldsymbol{s}_t, \boldsymbol{\psi}_m) = f(\boldsymbol{s}_t, \boldsymbol{\psi}_m) + \varepsilon_{f,t} \quad (4)$$

where $f(\boldsymbol{s}_t, \boldsymbol{\psi}_m)$ is the ideal equation of sensor network observation to robot, $\varepsilon_{f,t}$ is white Gaussian noise submitting to $N(0, \boldsymbol{Q}_t)$. From initial to *t*-th moment, sensor network observation on robot path can be expressed as $\boldsymbol{z}^{t,M} = \{\boldsymbol{z}_i^m\}_{i=1,\cdots,t;m=1,\cdots,M}$.

3 SIMULTANEOUS LOCALIZATION, CALIBRATION AND MAPPING

3.1 Fundamental thought

The fundamental thought of simultaneously localization, calibration and mapping is to estimate robot pose according to both robot controls, calibrated sensor network observations, and robot observations to environment features, simultaneously, to calibrate (and update) sensor network according to its observations to both robot and environment features, and to build (and update) the environment map according to both robot observations to features and sensor network observations to features.

From a probability perspective, the problem of simultaneous localization, calibration and mapping in network robot system can be expressed as a joint posterior distribution $p\left(\boldsymbol{\Psi}, s^t, \boldsymbol{\Theta}|\boldsymbol{u}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)$, that is, conditioned on robot control sequence $u^t = \{\boldsymbol{u}_i\}_{i=1,\cdots,t}$, robot observation sequence $\boldsymbol{y}^{t,N}$ of *N* environment features, sensor network observation sequence $\boldsymbol{z}^{t,M}$ from *M* sensor nodes to robot, and sensor network observation sequences $\boldsymbol{x}^{t,M,N}$ from *M* sensor nodes to *N* environment features, to estimate robot path $\boldsymbol{s}^t = \{\boldsymbol{s}_i\}_{i=1,\cdots,t}$, *M* sensor nodes location $\boldsymbol{\Psi} = \{\boldsymbol{\psi}_m\}_{m=1,\cdots,M}$, and the map $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_n\}_{n=1,\cdots,N}$ composed by *N* features. Based on Bayesian and Markov characteristics, the joint conditional distribution can be divided into:

$$\begin{aligned}
&p\left(\boldsymbol{\Psi}, \boldsymbol{s}^t, \boldsymbol{\Theta}|\boldsymbol{u}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)\\
&= p\left(\boldsymbol{\Psi}, \boldsymbol{\Theta}|\boldsymbol{s}^t, \boldsymbol{u}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right) \times\\
&\quad p\left(\boldsymbol{s}^t|\boldsymbol{u}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)\\
&= p\left(\boldsymbol{\Psi}, \boldsymbol{\Theta}|\boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right) p\left(\boldsymbol{s}^t|\boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)\\
&= p\left(\boldsymbol{\Theta}|\boldsymbol{\Psi}, \boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right) \times\\
&\quad p\left(\boldsymbol{\Psi}|\boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right) p\left(\boldsymbol{s}^t|\boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)\\
&= p\left(\boldsymbol{\Theta}|\boldsymbol{\Psi}, \boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}\right) \times\\
&\quad p\left(\boldsymbol{\Psi}|\boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{z}^{t,M}\right) p\left(\boldsymbol{s}^t|\boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)
\end{aligned} \quad (5)$$

Conditioned on the robot path, locations of sensor network nodes are independent, hence, decomposition can be obtained as follows:

$$\begin{aligned}
&p\left(\boldsymbol{\Psi}, \boldsymbol{s}^t, \boldsymbol{\Theta}|\boldsymbol{u}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)\\
&= p\left(\boldsymbol{\Theta}|\boldsymbol{\Psi}, \boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}\right) \times\\
&\quad \textstyle\prod_{m=1}^{M} p\left(\boldsymbol{\psi}_m|\boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{z}^{t,M}\right) p\left(\boldsymbol{s}^t|\boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)
\end{aligned} \quad (6)$$

Furthermore, conditioned on both robot path and sensor nodes locations, environment features locations are independent, and decomposition can be obtained as follows:

$$\begin{aligned}
&p\left(\boldsymbol{\Psi}, s^t, \boldsymbol{\Theta}|\boldsymbol{u}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)\\
&= \textstyle\prod_{n=1}^{N} p\left(\boldsymbol{\theta}_n|\boldsymbol{\Psi}, \boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}\right) \times\\
&\quad \textstyle\prod_{m=1}^{M} p\left(\boldsymbol{\psi}_m|\boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{z}^{t,M}\right) p\left(\boldsymbol{s}^t|\boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)
\end{aligned} \quad (7)$$

It can be seen from the above factorization that the calculation of the posterior over simultaneous localization, calibration and mapping in network robot system can be decomposed into an estimator over robot path $p\left(\boldsymbol{s}^t|\boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)$, *M* separate estimators over sensor node location parameters $p\left(\boldsymbol{\psi}_m|\boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{z}^{t,M}\right)$ and *N* separate estimators over environment feature location $p\left(\boldsymbol{\theta}_n|\boldsymbol{\Psi}, \boldsymbol{s}^t, \boldsymbol{x}^{t,M,N}, \boldsymbol{y}^{t,N}\right)$.

Based on framework of Rao-Blackwellized particle filtering, robot localization can be recursively implemented using particle filter, and both sensor network calibration and environment mapping of each particle can be implemented using Kalman filter.

In this paper we propose a solution using EPF for localization, and UKF for both calibration and mapping. For the *k*-th particle at *t*-th moment $\boldsymbol{S}_t^{(k)}$, the corresponding robot path can be expressed as $\boldsymbol{s}^{t,(k)}$, the corresponding location of the *m*-th sensor node can be expressed as $\boldsymbol{\psi}_t^{(k)(m)} = \left\{\boldsymbol{\mu}_{\psi,t}^{(k)(m)}, \Sigma_{\psi,t}^{(k)(m)}\right\}$ with mean $\boldsymbol{\mu}_{\psi,t}^{(k)(m)}$ and variance $\Sigma_{\psi,t}^{(k)(m)}$, and the corresponding location of the *n*-th environment feature can be expressed as $\boldsymbol{\theta}_t^{(k)(n)} = \left\{\boldsymbol{\mu}_{\theta,t}^{(k)(n)}, \Sigma_{\theta,t}^{(k)(n)}\right\}$ with mean $\boldsymbol{\mu}_{\theta,t}^{(k)(n)}$ and variance $\Sigma_{\theta,t}^{(k)(n)}$, hence, each particle possesses its own set of sensor nodes locations and environment features locations. So at *t*-th moment, the system state is characterized by set of *K* particles, where *k*-th particle $\boldsymbol{S}_t^{(k)}$ is defined as:

$$\boldsymbol{S}t^{(k)} = \left\{\boldsymbol{s}^{t,(k)}, \left\{\boldsymbol{\mu}_{\psi,t}^{(k)(m)}, \Sigma_{\psi,t}^{(k)(m)}\right\}_{m=1,\cdots,M}, \left\{\boldsymbol{\mu}_{\theta,t}^{(k)(n)}, \Sigma_{\theta,t}^{(k)(n)}\right\}_{n=1,\cdots,N}\right\} \tag{8}$$

And the reliability of the system can be expressed by *K* weighted particles as:

$$\boldsymbol{Bel}\left(X_t\right) = \left\{\boldsymbol{S}_t^{(k)}, w_t^{(k)}\right\}_{k=1,\cdots,K} \tag{9}$$

3.2 Robot localization

Theoretically, robot path particle $\boldsymbol{s}^{t,(k)}$ should be sampled from posterior distribution $p\left(\boldsymbol{s}_t|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)$ in order to approximate robot real path as close as possible. But actually it is hard to obtain this posterior. Based on Bayesian formula, we can divide this posterior into the following form:

$$\begin{aligned}
&\boldsymbol{s}^{t,(k)} \sim p\left(\boldsymbol{s}_t|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right) \\
&= \frac{p\left(\boldsymbol{z}_t^M|\boldsymbol{s}_t, \boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t-1,M}\right)}{p\left(\boldsymbol{z}_t^M|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t-1,M}\right)} \times \\
&\quad p\left(\boldsymbol{s}_t|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t-1,M}\right) \\
&= \frac{p\left(\boldsymbol{z}_t^M|\boldsymbol{s}_t, \boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t-1,M}\right)}{p\left(\boldsymbol{z}_t^M|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t-1,M}\right)} \times \\
&\quad \frac{p\left(\boldsymbol{y}_t^N|\boldsymbol{s}_t, \boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t-1,N}, \boldsymbol{z}^{t-1,M}\right)}{p\left(\boldsymbol{y}_t^N|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t-1,N}, \boldsymbol{z}^{t-1,M}\right)} \times \\
&\quad p\left(\boldsymbol{s}_t|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t-1,N}, \boldsymbol{z}^{t-1,M}\right) \\
&= \eta^{(k)} p\left(\boldsymbol{s}_t|\boldsymbol{s}_{t-1}^{(k)}, \boldsymbol{u}_t\right) \times \\
&\quad p\left(\boldsymbol{z}_t^M|\boldsymbol{s}_t, \boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t-1,M}\right) \times \\
&\quad p\left(\boldsymbol{y}_t^N|\boldsymbol{s}_t, \boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t-1,N}, \boldsymbol{z}^{t-1,M}\right) \\
&= \eta^{(k)} \underbrace{p\left(\boldsymbol{s}_t|\boldsymbol{s}_{t-1}^{(k)}, \boldsymbol{u}_t\right)}_{\sim N\left(\boldsymbol{s}_t; h\left(\boldsymbol{s}_{t-1}^{(k)}, \boldsymbol{u}_t\right), \boldsymbol{P}_t\right)} \times \\
&\quad \int \underbrace{p\left(\boldsymbol{z}_t^m|\boldsymbol{s}_t, \boldsymbol{\psi}_m\right)}_{\sim N\left(\boldsymbol{z}_t^m; f\left(\boldsymbol{s}_t, \boldsymbol{\psi}_m\right), \boldsymbol{Q}_t\right)} \underbrace{p\left(\boldsymbol{\psi}_m|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{z}^{t-1,m}\right)}_{\sim N\left(\boldsymbol{\psi}_m; \boldsymbol{\mu}_{\psi,t-1}^{(k)(m)}, \boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\right)} \mathrm{d}\boldsymbol{\psi}_m \times \\
&\quad \int \underbrace{p\left(\boldsymbol{y}_t^n|\boldsymbol{s}_t, \boldsymbol{\theta}_n\right)}_{\sim N\left(\boldsymbol{y}_t^n; g\left(\boldsymbol{\theta}_n, \boldsymbol{s}_t\right), \boldsymbol{R}_t\right)} \underbrace{p\left(\boldsymbol{\theta}_n|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{y}^{t-1,n}\right)}_{\sim N\left(\boldsymbol{\theta}_n; \boldsymbol{\mu}_{\theta,t-1}^{(k)(n)}, \boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\right)} \mathrm{d}\boldsymbol{\theta}_n \\
&= \eta^{(k)} p\left(\mathrm{s}_t|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t-1,N}, \boldsymbol{z}^{t-1,M}\right) \times \\
&\quad p\left(\boldsymbol{z}_t^M|\boldsymbol{s}_t, \boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t-1,M}\right) \times \\
&\quad p\left(\boldsymbol{y}_t^N|\boldsymbol{s}_t, \boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t-1,N}, \boldsymbol{z}^{t-1,M}\right)
\end{aligned} \tag{10}$$

where $\eta^{(k)}$ is the normalization factor, and

$$\begin{aligned}
\eta^{(k)} = p\left(\boldsymbol{z}_t^M|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t-1,M}\right)^{-1} \times \\
p\left(\boldsymbol{y}_t^N|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t-1,N}, \boldsymbol{z}^{t-1,M}\right)^{-1}
\end{aligned}$$

$p\left(\boldsymbol{s}_t|\boldsymbol{s}_{t-1}^{(k)}, \boldsymbol{u}_t\right)$, $p\left(\boldsymbol{z}_t^m|\boldsymbol{s}_t, \boldsymbol{\psi}_m\right)$ and $p\left(\boldsymbol{y}_t^n|\boldsymbol{s}_t, \boldsymbol{\theta}_n\right)$ are the robot motion model $h\left(\cdot\right)$, sensor observation model $f\left(\cdot\right)$ of the robot and robot observation model $g\left(\cdot\right)$ of the environment feature. $p\left(\boldsymbol{\psi}_m|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{z}^{t-1,m}\right)$, assumed as $N\left(\boldsymbol{\psi}_m; \boldsymbol{\mu}_{\psi,t-1}^{(k)(m)}, \boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\right)$, is the probability of obtaining *m*-th sensor location $\boldsymbol{\psi}_m$ according to robot path $\boldsymbol{s}^{t-1,(k)}$ and *m*-th sensor observations $\boldsymbol{z}^{t-1,m}$ on this path; $p\left(\boldsymbol{\theta}_n|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{y}^{t-1,n}\right)$, assumed as $N\left(\boldsymbol{\theta}_n; \boldsymbol{\mu}_{\theta,t-1}^{(k)(n)}, \boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\right)$, is the probability of obtain *n*-th environment feature location $\boldsymbol{\theta}_n$ according to robot path $s^{t-1,(k)}$ and the observations $\boldsymbol{y}^{t-1,n}$ to the *n*-th environment feature.

Because of nonlinearity of functions $h\left(\cdot\right)$, $f\left(\cdot\right)$ and $g\left(\cdot\right)$, it is still hard to obtain the numerical solution of (10). A direct method is to estimate (10) according to EKF, in which approximates these functions with first-degree Taylor expansion:

$$f\left(s_t, \boldsymbol{\psi}_m\right) \approx \hat{z}_t^{m,(k)} + \boldsymbol{F}_s\left(\boldsymbol{s}_t - \hat{\boldsymbol{s}}_t^{(k)}\right) + \boldsymbol{F}_\psi\left(\boldsymbol{\psi}_m - \boldsymbol{\mu}_{\psi,t-1}^{(k)(m)}\right) \tag{11a}$$

$$g\left(\boldsymbol{\theta}_n, \boldsymbol{s}_t\right) \approx \hat{\boldsymbol{y}}_t^{n,(k)} + \boldsymbol{G}_s\left(\boldsymbol{s}_t - \hat{\boldsymbol{s}}_t^{(k)}\right) + \boldsymbol{G}_\theta\left(\boldsymbol{\theta}_n - \boldsymbol{\mu}_{\theta,t-1}^{(k)(n)}\right) \tag{11b}$$

where $\hat{s}_t^{(k)} = h\left(s_{t-1}^{(k)}, \boldsymbol{u}_t\right)$ is the predicted robot pose according to robot motion model $h(\cdot)$, $\hat{z}_t^{m,(k)} = f\left(\hat{s}_t^{(k)}, \boldsymbol{\mu}_{\psi,t-1}^{(k)(m)}\right)$ is the predicted sensor network observation to robot according to observation model $f(\cdot)$, $\hat{\boldsymbol{y}}_t^{n,(k)} = g\left(\hat{\boldsymbol{s}}_t^{(k)}, \boldsymbol{\mu}_{\theta,t-1}^{(k)(n)}\right)$ is the predicted robot observation to environment feature according to observation model $g(\cdot)$. Matrix $\boldsymbol{F}_s$ and $\boldsymbol{F}_\psi$ are the Jacobians of $f(\cdot)$, respectively. Matrix $\boldsymbol{G}_s$ and $\boldsymbol{G}_\theta$ are the derivatives of $g(\cdot)$ with respect to $\boldsymbol{s}_t$ and $\boldsymbol{\theta}_n$. According to the approximation, the two integral terms can be transformed into two Gaussian distributions:

$$\begin{aligned}&\int p\left(\boldsymbol{z}_t^m|\boldsymbol{s}_t, \boldsymbol{\psi}_m\right) p\left(\boldsymbol{\psi}_m|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{z}^{t-1,m}\right) \mathrm{d}\boldsymbol{\psi}_m \sim \\ &\quad N\left(\boldsymbol{z}_t^m; \hat{\boldsymbol{z}}_t^{m,(k)} + \boldsymbol{F}_s\boldsymbol{s}_t - \boldsymbol{F}_s\hat{\boldsymbol{s}}_t^{(k)}, \boldsymbol{Q}_t + \boldsymbol{F}_\psi\boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\boldsymbol{F}_\psi^{\mathrm{T}}\right)\end{aligned} \tag{12a}$$

$$\begin{aligned}&\int p\left(y_t^n|\boldsymbol{s}_t, \boldsymbol{\theta}_n\right) p\left(\boldsymbol{\theta}_n|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{y}^{t-1,n}\right) \mathrm{d}\boldsymbol{\theta}_n \sim \\ &\quad N\left(\boldsymbol{y}_t^n; \hat{\boldsymbol{y}}_t^{n,(k)} + \boldsymbol{G}_s\boldsymbol{s}_t - \boldsymbol{G}_s\hat{\boldsymbol{s}}_t^{(k)}, \boldsymbol{R}_t + \boldsymbol{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\boldsymbol{G}_\theta^{\mathrm{T}}\right)\end{aligned} \tag{12b}$$

And then the posterior (10) can be transformed into product of several Gaussian distributions denoted as:

$$p\left(\boldsymbol{s}_t|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right) = \eta' \exp\left\{-\boldsymbol{E}_t^{(k)}\right\} \tag{13}$$

Where

$$\begin{aligned}\boldsymbol{E}_t^{(k)} = &\tfrac{1}{2}\left(\boldsymbol{s}_t - \hat{\boldsymbol{s}}_t^{(k)}\right)^{\mathrm{T}} \boldsymbol{P}_t^{-1}\left(\boldsymbol{s}_t - \hat{\boldsymbol{s}}_t^{(k)}\right) + \\ &\tfrac{1}{2}\left(\boldsymbol{z}_t^m - \hat{\boldsymbol{z}}_t^{m,(k)} - \boldsymbol{F}_s\boldsymbol{s}_t + \boldsymbol{F}_s\hat{\boldsymbol{s}}_t^{(k)}\right)^{\mathrm{T}} \times \\ &\left(\boldsymbol{Q}_t + \boldsymbol{F}_\psi\boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\boldsymbol{F}_\psi^{\mathrm{T}}\right)^{-1}\left(\mathbf{z}_t^m - \hat{\mathbf{z}}_t^{m,(k)} - \mathbf{F}_s\mathbf{s}_t + \mathbf{F}_s\hat{\mathbf{s}}_t^{(k)}\right) + \\ &\tfrac{1}{2}\left(\mathbf{y}_t^n - \hat{\mathbf{y}}_t^{n,(k)} - \mathbf{G}_s\mathbf{s}_t + \mathbf{G}_s\hat{\mathbf{s}}_t^{(k)}\right)^{\mathrm{T}} \times \\ &\left(\mathbf{R}_t + \mathbf{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\mathbf{G}_\theta^{\mathrm{T}}\right)^{-1}\left(\mathbf{y}_t^n - \hat{\mathbf{y}}_t^{n,(k)} - \mathbf{G}_s\mathbf{s}_t + \mathbf{G}_s\hat{\mathbf{s}}_t^{(k)}\right)\end{aligned} \tag{14}$$

It can be seen obviously that, $\boldsymbol{E}_t^{(k)}$ is quadratic in robot pose $\boldsymbol{s}_t$, hence $p\left(\boldsymbol{s}_t|\boldsymbol{s}^{t-1,(k)}, \boldsymbol{u}^t, \boldsymbol{y}^{t,N}, \boldsymbol{z}^{t,M}\right)$ is still Gaussian, whose mean and covariance are equivalent to the minimum of $\boldsymbol{E}_t^{(k)}$ and its curvature.

The first and second derivatives of $\boldsymbol{E}_t^{(k)}$ with respect to $\boldsymbol{s}_t$ are:

$$\begin{aligned}\frac{\partial\mathbf{E}_t^{(k)}}{\partial\mathbf{s}_t} = &\mathbf{P}_t^{-1}\left(\mathbf{s}_t - \hat{\mathbf{s}}_t^{(k)}\right) - \\ &\mathbf{F}_s^{\mathrm{T}}\left(\mathbf{Q}_t + \mathbf{F}_\psi\boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\mathbf{F}_\psi^{\mathrm{T}}\right)^{-1}\left(\mathbf{z}_t^m - \hat{\mathbf{z}}_t^{m,(k)} - \mathbf{F}_s\mathbf{s}_t + \mathbf{F}_s\hat{\mathbf{s}}_t^{(k)}\right) - \\ &\mathbf{G}_s^{\mathrm{T}}\left(\mathbf{R}_t + \mathbf{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\mathbf{G}_\theta^{\mathrm{T}}\right)^{-1}\left(\mathbf{y}_t^n - \hat{\mathbf{y}}_t^{n,(k)} - \mathbf{G}_s\mathbf{s}_t + \mathbf{G}_s\hat{\mathbf{s}}_t^{(k)}\right)\end{aligned} \tag{15}$$

$$\begin{aligned}\frac{\partial^2\mathbf{E}_t^{(k)}}{\partial\mathbf{s}_t^2} = &\mathbf{P}_t^{-1} + \mathbf{F}_s^{\mathrm{T}}\left(\mathbf{Q}_t + \mathbf{F}_\psi\boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\mathbf{F}_\psi^{\mathrm{T}}\right)^{-1}\mathbf{F}+ \\ &\mathbf{G}_s^{\mathrm{T}}\left(\mathbf{R}_t + \mathbf{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\mathbf{G}_\theta^{\mathrm{T}}\right)^{-1}\mathbf{G}_s\end{aligned} \tag{16}$$

The covariance $\boldsymbol{\Sigma}_{s,t}^{(k)}$ of $\mathbf{s}_t$ is now obtained by the inverse of the second derivative:

$$\begin{aligned}\boldsymbol{\Sigma}_{s,t}^{(k)} = \Bigg[&\mathbf{P}_t^{-1} + \mathbf{F}_s^{\mathrm{T}}\left(\mathbf{Q}_t + \mathbf{F}_\psi\boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\mathbf{F}_\psi^{\mathrm{T}}\right)^{-1}\mathbf{F}+ \\ &\mathbf{G}_s^{\mathrm{T}}\left(\mathbf{R}_t + \mathbf{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\mathbf{G}_\theta^{\mathrm{T}}\right)^{-1}\mathbf{G}_s\Bigg]^{-1}\end{aligned} \tag{17}$$

The mean $\boldsymbol{\mu}_{s,t}^{(k)}$ of $\mathbf{s}_t$ is obtained by setting the first derivative to zero, which is:

$$\begin{aligned}\boldsymbol{\mu}_{s,t}^{(k)} &= \hat{\mathbf{s}}_t^{(k)} + \boldsymbol{\Sigma}_{s,t}^{(k)}\mathbf{F}_s^{\mathrm{T}}\left(\mathbf{Q}_t + \mathbf{F}_\psi\boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\mathbf{F}_\psi^{\mathrm{T}}\right)^{-1}\left(\mathbf{z}_t^m - \hat{\mathbf{z}}_t^{m,(k)}\right) \\ &+\boldsymbol{\Sigma}_{s,t}^{(k)}\mathbf{G}_s^{\mathrm{T}}\left(\mathbf{R}_t + \mathbf{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\mathbf{G}_\theta^{\mathrm{T}}\right)^{-1}\left(\mathbf{y}_t^n - \hat{\mathbf{y}}_t^{n,(k)}\right)\end{aligned} \tag{18}$$

So, according to the robot controls $\mathbf{u}^t$, sensor network observations to robot $\mathbf{z}^{t,M}$, and robot observations to localized environment feature $\mathbf{y}^{t,N}$ we obtain the proposal distribution $p\left(\mathbf{s}_t|\mathbf{s}^{t-1,(k)}, \mathbf{u}^t, \mathbf{y}^{t,N}, \mathbf{z}^{t,M}\right)$ with Gaussian form $N\left(\mathbf{s}_t; \mu_{s,t}^{(k)}, \boldsymbol{\Sigma}_{s,t}^{(k)}\right)$, from which new particles are sampled.

Up to now, the solution to robot localization is given utilizing EPF [25]. Besides, it is easy to find that, when there are only robot observations to environment features but no sensor network observations to robot, that is, $\{\mathbf{z}_t\} =$ NULL and $\{\mathbf{y}_t\} \neq$ NULL, then (17), (18) can be rewritten into:

$$\boldsymbol{\Sigma}_{s,t}^{(k)} = \left[\mathbf{P}_t^{-1} + \mathbf{G}_s^{\mathrm{T}}\left(\mathbf{R}_t + \mathbf{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\mathbf{G}_\theta^{\mathrm{T}}\right)^{-1}\mathbf{G}_s\right]^{-1}$$

$$\boldsymbol{\mu}_{s,t}^{(k)} = \hat{\mathbf{s}}_t^{(k)} + \boldsymbol{\Sigma}_{s,t}^{(k)}\mathbf{G}_s^{\mathrm{T}}\left(\mathbf{R}_t + \mathbf{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\mathbf{G}_\theta^{\mathrm{T}}\right)^{-1}\left(\mathbf{y}_t^n - \hat{\mathbf{y}}_t^{n,(k)}\right)$$

In which case, the solution degenerates into the localization of FastSLAM algorithm.

Because of the deduce phenomenon, the robot path simulated by particles biases from the true path. To deal with this problem, the resample step is involved. The differences between normalizing factors $\eta^{(k)}$ are ignored in (10), and in order to remedy it we choose the ratio between true posterior distribution and proposal distribution as particle weight, which is:

$$w_t^{(k)} = \frac{\text{true}}{\text{proposal}} = \frac{p\left(\mathbf{s}^{t,(k)}|\mathbf{u}^t, \mathbf{y}^{t,N}, \mathbf{z}^{t,M}\right)}{p\left(\mathbf{s}^{t-1,(k)}|\mathbf{u}^{t-1}, \mathbf{y}^{t-1,N}, \mathbf{z}^{t-1,M}\right) p\left(\mathbf{s}_t^{(k)}|\mathbf{u}^t, \mathbf{s}^{t-1,(k)}, \mathbf{y}^{t,N}, \mathbf{z}^{t,M}\right)} \tag{19}$$

Depending on Bayesian formula and Markov characteristics we can obtain the following factorization:

$$\begin{aligned}
&w_t^{(k)} = \\
&\frac{p\left(\mathbf{s}_t^{(k)}|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t,M}\right)p\left(\mathbf{s}^{t-1,(k)}|\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t,M}\right)}{p\left(\mathbf{s}^{t-1,(k)}|\mathbf{u}^{t-1},\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right)p\left(\mathbf{s}_t^{(k)}|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t,M}\right)} \\
&= \frac{p\left(\mathbf{z}_t^M|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t-1,M}\right)p\left(\mathbf{s}^{t-1,(k)}|\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t-1,M}\right)}{p\left(\mathbf{z}_t^M|\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t-1,M}\right)p\left(\mathbf{s}^{t-1,(k)}|\mathbf{u}^{t-1},\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right)} \\
&= \frac{p\left(\mathbf{y}_t^N|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right)}{p\left(\mathbf{y}_t^N|\mathbf{u}^t,\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right)}\times \\
&\frac{p\left(\mathbf{z}_t^M|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t-1,M}\right)p\left(\mathbf{s}^{t-1,(k)}|\mathbf{u}^t,\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right)}{p\left(\mathbf{z}_t^M|\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t-1,M}\right)p\left(\mathbf{s}^{t-1,(k)}|\mathbf{u}^{t-1},\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right)} \\
&\propto p\left(\mathbf{z}_t^M|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t-1,M}\right)\times \\
&\quad p\left(\mathbf{y}_t^N|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right)
\end{aligned} \tag{20}$$

It can be seen that, particle weight is the inverse of normalizing factors $\eta^{(k)}$ in (10). In the above factorization, former distribution $p\left(\mathbf{z}_t^M|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t-1,M}\right)$ can be decomposed further as:

$$\begin{aligned}
&p\left(\mathbf{z}_t^M|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t-1,M}\right) \\
&= \int p\left(\mathbf{z}_t^M|\mathbf{s}_t,\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{z}^{t-1,M}\right)p\left(\mathbf{s}_t|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{z}^{t-1},\mathbf{y}^{t,N}\right)\mathrm{d}\mathbf{s}_t \\
&= \int p\left(\mathbf{z}_t^M|\mathbf{s}_t,\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{z}^{t-1,M}\right)p\left(\mathbf{s}_t|\mathbf{s}_{t-1}^{(k)},\mathbf{u}_t\right)\mathbf{ds}_t \\
&= \iint p\left(\mathbf{z}_t^m|\boldsymbol{\psi}_m,\mathbf{s}_t\right)p\left(\boldsymbol{\psi}_m|\mathbf{s}^{t-1,(k)},\mathbf{z}^{t-1,m}\right)\mathrm{d}\boldsymbol{\psi}_m p\left(\mathbf{s}_t|\mathbf{s}_{t-1}^{(k)},\mathbf{u}_t\right)\mathrm{d}\boldsymbol{s}_t
\end{aligned} \tag{21}$$

In the same way, the latter distribution $p\left(\mathbf{y}_t^N|\mathbf{s}^{t-1},\mathbf{u}^t,\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right)$ can be decomposed as:

$$\begin{aligned}
&p\left(\mathbf{y}_t^N|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right) = \\
&\iint p\left(\mathbf{y}_t^n|\mathbf{s}_t,\boldsymbol{\theta}_n\right)p\left(\boldsymbol{\theta}_n|\mathbf{s}^{t-1,(k)},\mathbf{y}^{t-1,n}\right)\mathrm{d}\boldsymbol{\theta}_n p\left(\mathbf{s}_t|\mathbf{s}_{t-1}^{(k)},\mathbf{u}_t\right)\mathrm{d}\boldsymbol{s}_t
\end{aligned} \tag{22}$$

Then the weight can be denoted as product of two integrals:

$$\begin{aligned}
&w_t^{(k)} \propto p\left(\mathbf{z}_t^M|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t,N},\mathbf{z}^{t-1,M}\right)\times \\
&\quad p\left(\mathbf{y}_t^N|\mathbf{s}^{t-1,(k)},\mathbf{u}^t,\mathbf{y}^{t-1,N},\mathbf{z}^{t-1,M}\right) = \\
&\iint \underbrace{p\left(\mathbf{z}_t^m|\mathbf{s}_t,\boldsymbol{\psi}_m\right)}_{\sim N\left(\mathbf{z}_t^m;f(\mathbf{s}_t,\boldsymbol{\psi}_m),\mathbf{Q}_t\right)} \underbrace{p\left(\boldsymbol{\psi}_m|\mathbf{s}^{t-1},\mathbf{z}^{t-1,m}\right)}_{\sim N\left(\boldsymbol{\psi}_m;\boldsymbol{\mu}_{\psi,t-1}^{(k)(m)},\boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\right)} \mathrm{d}\boldsymbol{\psi}_m \\
&\underbrace{p\left(\mathbf{s}_t|\mathbf{s}_{t-1}^{(k)},\mathbf{u}_t\right)}_{\sim N\left(\mathbf{s}_t;h\left(\mathbf{s}_{t-1}^{(k)},\mathbf{u}_t\right),\mathbf{P}_t\right)} \mathrm{d}\boldsymbol{s}_t \times \iint \underbrace{p\left(\mathbf{y}_t^n|\mathbf{s}_t,\boldsymbol{\theta}_n\right)}_{\sim N\left(\mathbf{y}_t^n;g(\boldsymbol{\theta}_n,\mathbf{s}_t),\mathbf{R}_t\right)} \\
&\underbrace{p\left(\boldsymbol{\theta}_n|\mathbf{s}^{t-1},\mathbf{y}^{t-1,n}\right)}_{\sim N\left(\boldsymbol{\theta}_n;\boldsymbol{\mu}_{\theta,t-1}^{(k)(n)},\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\right)} \mathrm{d}\boldsymbol{\theta}_n \underbrace{p\left(\mathbf{s}_t|\mathbf{s}_{t-1}^{(k)},\mathbf{u}_t\right)}_{\sim N\left(\mathbf{s}_t;h\left(\mathbf{s}_{t-1}^{(k)},\mathbf{u}_t\right),\mathbf{P}_t\right)} \mathrm{d}\boldsymbol{s}_t
\end{aligned} \tag{23}$$

According to the approximation like (12a), we can obtain their Gaussian forms as follows:

$$\begin{aligned}
&\iint p\left(\mathbf{z}_t^m|\mathbf{s}_t,\boldsymbol{\psi}_m\right)p\left(\boldsymbol{\psi}_m|\mathbf{s}^{t-1},\mathbf{z}^{t-1,m}\right)\mathrm{d}\boldsymbol{\psi}_m p\left(\mathbf{s}_t|\mathbf{s}_{t-1}^{(k)},\mathbf{u}_t\right)\mathrm{d}\boldsymbol{s}_t \\
&\quad \sim N\left(\mathbf{z}_t^m;\hat{\mathbf{z}}_t^{(k)},\mathbf{Q}_t+\mathbf{F}_s\mathbf{P}_t\mathbf{F}_s^{\mathbf{T}}+\mathbf{F}_\psi\boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\mathbf{F}_\psi^{\mathbf{T}}\right)
\end{aligned}$$

$$\begin{aligned}
&\iint p\left(\mathbf{y}_t^n|\mathbf{s}_t,\boldsymbol{\theta}_n\right)p\left(\boldsymbol{\theta}_n|\mathbf{s}^{t-1},\mathbf{y}^{t-1,n}\right)\mathrm{d}\boldsymbol{\theta}_n p\left(\mathbf{s}_t|\mathbf{s}_{t-1}^{(k)},\mathbf{u}_t\right)\mathrm{d}\boldsymbol{s}_t \\
&\quad \sim N\left(\mathbf{y}_t^n;\hat{\mathbf{y}}_t^{(k)},\mathbf{R}_t+\mathbf{G}_s\mathbf{P}_t\mathbf{G}_s^{\mathbf{T}}+\mathbf{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\mathbf{F}_\theta^{\mathbf{T}}\right)
\end{aligned}$$

Assume that $\mathbf{L}_{z,t}^{(k)} = \mathbf{Q}_t+\mathbf{F}_s\mathbf{P}_t\mathbf{F}_s^{\mathbf{T}}+\mathbf{F}_\psi\boldsymbol{\Sigma}_{\psi,t-1}^{(k)(m)}\mathbf{F}_\psi^{\mathbf{T}}$, $\mathbf{L}_{y,t}^{(k)} = \mathbf{R}_t+\mathbf{G}_s\mathbf{P}_t\mathbf{G}_s^{\mathbf{T}}+\mathbf{G}_\theta\boldsymbol{\Sigma}_{\theta,t-1}^{(k)(n)}\mathbf{F}_\theta^{\mathbf{T}}$, and the weight of *k*-th particle can be obtained as:

$$\begin{aligned}
w_t^{(k)} = &\left|2\pi\mathbf{L}_{y,t}^{(k)}\right|^{-1/2}\left|2\pi\mathbf{L}_{z,t}^{(k)}\right|^{-1/2}\times \\
&\exp\left\{-\tfrac{1}{2}\left(\mathbf{z}_t^m-\hat{\mathbf{z}}_t^{m,(k)}\right)^{\mathrm{T}}\mathbf{L}_{z,t}^{(i)-1}\left(\mathbf{z}_t^m-\hat{\mathbf{z}}_t^{m,(k)}\right)-\right. \\
&\left.\tfrac{1}{2}\left(\mathbf{y}_t^n-\hat{\mathbf{y}}_t^{n,(k)}\right)^{\mathrm{T}}\mathbf{L}_{y,t}^{(k)}\left(\mathbf{y}_t^n-\hat{\mathbf{y}}_t^{n,(k)}\right)\right\}
\end{aligned} \tag{24}$$

The weights should be normalized and particles are resampled according to the normalized weights. Here, we calculate an effective particle number $N_{eff} = 1\Big/\sum_{k=1}^{K}\left(w_t^{(k)}\right)^2$. When N_{eff} is smaller than a threshold value (0.7K in this article), the weighted particles set $\left\{\mathbf{S}_t^{(k)},w_t^{(k)}\right\}_{k=1,\cdots,K}$ would be transformed into a new particle set $\left\{\mathbf{S}_t^{(k)},1/K\right\}_{k=1,\cdots,K}$ with equal weights.

3.3 Sensor network calibration

Depend on Bayesian formula and Markov characteristics, it can be obtained that:

$$\begin{aligned}
&p\left(\boldsymbol{\psi}_m|\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{z}^{t,M}\right) \\
&= \frac{p\left(\mathbf{z}_t^M|\boldsymbol{\psi}_m,\mathbf{x}^{t,M,N},\mathbf{z}^{t-1,M}\right)}{p\left(\mathbf{z}_t^M|\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{z}^{t-1,M}\right)}p\left(\boldsymbol{\psi}_m|\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{z}^{t-1,M}\right) \\
&= \frac{p\left(\mathbf{z}_t^M|\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{z}^{t-1,M}\right)}{p\left(\mathbf{z}_t^M|\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{z}^{t-1,M}\right)}\times \\
&\quad \frac{p\left(\mathbf{x}_t^{M,N}|\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t-1,M,N},\mathbf{z}^{t-1,M}\right)}{p\left(\mathbf{x}_t^{M,N}|\mathbf{s}^t,\mathbf{x}^{t-1,M,N},\mathbf{z}^{t-1,M}\right)}\times \\
&\quad p\left(\boldsymbol{\psi}_m|\mathbf{s}^t,\mathbf{x}^{t-1,M,N},\mathbf{z}^{t-1,M}\right) \\
&= \eta p\left(\mathbf{z}_t^M|\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{z}^{t-1,M}\right)\times \\
&\quad p\left(\mathbf{x}_t^{M,N}|\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t-1,M,N},\mathbf{z}^{t-1,M}\right)\times \\
&\quad p\left(\boldsymbol{\psi}_m|\mathbf{s}^t,\mathbf{x}^{t-1,M,N},\mathbf{z}^{t-1,M}\right) = \eta \underbrace{p\left(\mathbf{z}_t^M|\mathbf{s}_t,\psi_m\right)}_{\sim N\left(\mathbf{z}_t^M;f(\mathbf{s}_t,\boldsymbol{\psi}_m),\mathbf{Q}_t\right)} \times \\
&\quad \int \underbrace{p\left(\mathbf{x}_t^{M,N}|\boldsymbol{\psi}_m,\boldsymbol{\theta}_n\right)}_{\sim N\left(\mathbf{x}_t^{M,N};d(\boldsymbol{\psi}_m,\boldsymbol{\theta}_n),\mathbf{T}_t\right)} \underbrace{p\left(\boldsymbol{\theta}_n|\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t-1,M,N}\right)}_{\sim N\left(\boldsymbol{\theta}_n;\boldsymbol{\mu}_{\theta,t-1}^{(k)},\boldsymbol{\Sigma}_{\theta,t-1}^{(k)}\right)} \mathbf{d}\boldsymbol{\theta}_n \times \\
&\quad \underbrace{p\left(\boldsymbol{\psi}_m|\mathbf{s}^{t-1},\mathbf{x}^{t-1,M,N},\mathbf{z}^{t-1,M}\right)}_{\sim N\left(\boldsymbol{\psi}_m;\boldsymbol{\mu}_{\psi,t-1}^{(k)},\boldsymbol{\Sigma}_{\psi,t-1}^{(k)}\right)}
\end{aligned} \tag{25}$$

where $p\left(\mathbf{z}_t^M|\mathbf{s}_t,\psi_m\right)$ and $p\left(\mathbf{x}_t^{M,N}|\psi_m,\boldsymbol{\theta}_n\right)$ denote the sensor network observation to robot and sensor network observation to environment features. Because they are nonlinear, it is hard to obtain the numerical solution of the proposal distribution $p\left(\boldsymbol{\psi}_m|\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{z}^{t,M}\right)$.

Here UKF algorithm [26, 27] is used recursively to estimate the sensor node location at present moment based on mean $\boldsymbol{\mu}_{\psi,t-1}^{(k)}$ and variance $\boldsymbol{\Sigma}_{\psi,t-1}^{(k)}$ estimated at (t-1)-th moment, the main algorithm is as following:

Step 1 (Calculate Sigma points):
Calculate $2n_\psi+1$ Sigma points $\left\{\xi_{t-1}^{(k),(i)}\right\}_{i=0,1,\ldots,2n_\psi}$ and weights according to mean $\boldsymbol{\mu}_{\psi,t-1}^{(k)}$ and variance $\boldsymbol{\Sigma}_{\psi,t-1}^{(k)}$:

$$\xi_{t-1}^{(k),(0)}=\boldsymbol{\mu}_{\psi,t-1}^{(k)}$$

$$\xi_{t-1}^{(k),(i)}=\boldsymbol{\mu}_{\psi,t-1}^{(k)}+\left(\sqrt{(n_\psi+\lambda)\boldsymbol{\Sigma}_{\psi,t-1}^{(k)}}\right)_i \quad i=1,2,\ldots,n_\psi$$

$$\xi_{t-1}^{(k),(i)}=\boldsymbol{\mu}_{\psi,t-1}^{(k)}-\left(\sqrt{(n_\psi+\lambda)\boldsymbol{\Sigma}_{\psi,t-1}^{(k)}}\right)_i \quad i=n_\psi+1,n_\psi+2,\ldots,2n_\psi$$

$$w_0^{(k),(m)}=\frac{\lambda}{n_\psi+\lambda}$$

$$w_0^{(k),(c)}=\frac{\lambda}{n_\psi+\lambda}+\left(1-\alpha^2+\beta\right)$$

$$w_i^{(k),(m)}=w_i^{(k),(c)}=\frac{1}{2\left(n_\psi+\lambda\right)} i=1,2,\ldots,2n_\psi$$

where n_ψ is the dimension of sensor location, $\lambda=\alpha^2\left(n_\psi+\kappa\right)-n_\psi$, here we also set α as 0.01, κ as 0, and β as 2. $\left(\sqrt{(n_\psi+\lambda)\boldsymbol{\Sigma}_{\psi,t-1}^{(k)}}\right)_i$ is the i-th column of matrix square root of $(n_\psi+\lambda)\boldsymbol{\Sigma}_{\psi,t-1}^{(k)}$. $w_{i=0,1,\ldots,2n_\psi}^{(k),(m)}$ is the weight of first-degree peculiarity, $w_{i=0,1,\ldots,2n_\psi}^{(k),(c)}$ is the weight of second-degree peculiarity.

Step 2 (Time update):
Transform Sigma points into $\left\{\xi_{t|t-1}^{(k),(i)}\right\}_{i=0,1,\ldots,2n_\psi}$, and calculate the predicted mean $\hat{\boldsymbol{\mu}}_{\psi,t|t-1}^{(k)}$ and variance $\hat{\boldsymbol{\Sigma}}_{\psi,t|t-1}^{(k)}$:

$$\xi_{t|t-1}^{(k),(i)}=\xi_{t-1}^{(k),(i)} i=0,1,\ldots,2n_\psi$$

$$\hat{\boldsymbol{\mu}}_{\psi,t|t-1}^{(k)}=\sum_{i=0}^{2n_\psi}w_i^{(k),(m)}\xi_{t|t-1}^{(k),(i)}$$

$$\hat{\boldsymbol{\Sigma}}_{\psi,t|t-1}^{(k)}=\sum_{i=0}^{2n_\psi}w_i^{(k),(c)}\left(\xi_{t|t-1}^{(k),(i)}-\hat{\boldsymbol{\mu}}_{\psi,t|t-1}^{(k)}\right)\left(\xi_{t|t-1}^{(k),(i)}-\hat{\boldsymbol{\mu}}_{\psi,t|t-1}^{(k)}\right)^{\mathrm{T}}$$

Step 3 (Update with measurement $\boldsymbol{z}_t$):

i) Calculate the predicted measurement $\hat{\mathbf{z}}_t^{(\mathbf{k})}$

$$\zeta_{t|t-1}^{(k),(i)}=f\left(\mathbf{s}_t^{(k)},\xi_{t|t-1}^{(k),(i)}\right) i=0,1,\ldots,2n_\psi$$

$$\hat{\mathbf{z}}_t^{(k)}=\sum_{i=0}^{2n_\psi}w_i^{(k),(m)}\zeta_{t|t-1}^{(k),(i)}$$

ii) Update with true measurement $\mathbf{z}_t$

$$\mathbf{P}_{\hat{z}_t}^{(k)}=\sum_{i=0}^{2n_\psi}w_i^{(k),(c)}\left(\zeta_{t|t-1}^{(k),(i)}-\hat{\boldsymbol{z}}_t^{(k)}\right)\left(\zeta_{t|t-1}^{(k),(i)}-\hat{\mathbf{z}}_t^{(k)}\right)^{\mathrm{T}}+\mathbf{Q}_t$$

$$\mathbf{P}_{\hat{\psi}_t\hat{z}_t}^{(k)}=\sum_{i=0}^{2n_\psi}w_i^{(k),(c)}\left(\xi_{t|t-1}^{(k),(i)}-\hat{\boldsymbol{\mu}}_{\psi,t|t-1}^{(k)}\right)\left(\zeta_{t|t-1}^{(k),(i)}-\hat{\mathbf{z}}_t^{(k)}\right)^{\mathrm{T}}$$

$$\hat{\boldsymbol{\mu}}_{\psi,t|t}^{(k)}=\hat{\boldsymbol{\mu}}_{\psi,t|t-1}^{(k)}+\mathbf{K}_t\left(\mathbf{z}_t-\hat{\mathbf{z}}_t^{(k)}\right)$$

$$\mathbf{K}_t=\mathbf{P}_{\hat{\psi}_t\hat{z}_t}^{(k)}\left(\mathbf{P}_{\hat{z}_t}^{(k)}\right)^{-1}$$

$$\hat{\boldsymbol{\Sigma}}_{\psi,t|t}^{(k)}=\hat{\boldsymbol{\Sigma}}_{\psi,t|t-1}^{(k)}-\mathbf{K}_t\mathbf{P}_{\hat{z}_t}^{(k)}\mathbf{K}_t^{\mathrm{T}}$$

Step 4 (Update with measurement $\mathbf{x}_t$):

i) Calculate the predicted measurement $\hat{\mathbf{x}}_t^{(\mathbf{k})}$

$$\varsigma_t^{(k),(i)}=d\left(\boldsymbol{\theta}_n,\xi_t^{(k),(i)}\right) i=0,1,\ldots,2n_\psi$$

$$\hat{\mathbf{x}}_t^{(k)}=\sum_{i=0}^{2n_\psi}w_i^{(k),(m)}\varsigma_t^{(k),(i)}$$

ii) Update with true measurement $\mathbf{x}_t$

$$\mathbf{P}_{\hat{x}_t}^{(k)}=\sum_{i=0}^{2n_\psi}w_i^{(k),(c)}\left(\varsigma_{t|t-1}^{(k),(i)}-\hat{\boldsymbol{x}}_t^{(k)}\right)\left(\varsigma_t^{(k),(i)}-\hat{\mathbf{x}}_t^{(k)}\right)^{\mathrm{T}}+\mathbf{T}_t$$

$$\mathbf{P}_{\hat{\psi}_t\hat{x}_t}^{(k)}=\sum_{i=0}^{2n_\psi}w_i^{(k),(c)}\left(\xi_{t|t-1}^{(k),(i)}-\hat{\boldsymbol{\mu}}_{\psi,t|t-1}^{(k)}\right)\left(\varsigma_t^{(k),(i)}-\hat{\mathbf{x}}_t^{(k)}\right)^{\mathrm{T}}$$

$$\boldsymbol{\mu}_{\psi,t}^{(k)}=\hat{\boldsymbol{\mu}}_{\psi,t|t}^{(k)}+\mathbf{K}_t\left(\mathbf{x}_t-\hat{\mathbf{x}}_t^{(k)}\right)$$

$$\mathbf{K}_t=\mathbf{P}_{\hat{\psi}_t\hat{x}_t}^{(k)}\left(\mathbf{P}_{\hat{x}_t}^{(k)}\right)^{-1}$$

$$\boldsymbol{\Sigma}_{\psi,t}^{(k)}=\hat{\boldsymbol{\Sigma}}_{\psi,t|t}^{(k)}-\mathbf{K}_t\mathbf{P}_{\hat{x}_t}^{(k)}\mathbf{K}_t^{\mathrm{T}}$$

So, here we obtain mean $\boldsymbol{\mu}_{\psi,t}^{(k)}$ and variance $\boldsymbol{\Sigma}_{\psi,t}^{(k)}$ of sensor at tth moment based on UKF.

3.4 Environment mapping

Similar to formula (25), it can be known depend on Bayesian formula and Markov characteristics:

$$
\begin{aligned}
& p\left(\boldsymbol{\theta}_n|\boldsymbol{\psi}_m, \mathbf{s}^t, \mathbf{x}^{t,M,N}, \mathbf{y}^{t,N}\right) \\
= & \frac{p\left(\mathbf{y}_t^N|\boldsymbol{\theta}_n,\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{y}^{t-1,N}\right)}{p\left(\mathbf{y}_t^N|\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{y}^{t-1,N}\right)} \times \\
& p\left(\boldsymbol{\theta}_n|\boldsymbol{\psi}_m, \mathbf{s}^t, \mathbf{x}^{t,M,N}, \mathbf{y}^{t-1,N}\right) \\
= & \frac{p\left(\mathbf{y}_t^N|\boldsymbol{\theta}_n,\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{y}^{t-1,N}\right)}{p\left(\mathbf{y}_t^N|\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t,M,N},\mathbf{y}^{t-1,N}\right)} \times \\
& \frac{p\left(\mathbf{x}_t^{M,N}|\boldsymbol{\theta}_n,\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t-1,M,N},\mathbf{y}^{t-1,N}\right)}{p\left(\mathbf{x}_t^{M,N}|\boldsymbol{\psi}_m,\mathbf{s}^t,\mathbf{x}^{t-1,M,N},\mathbf{y}^{t-1,N}\right)} \times \\
& p\left(\boldsymbol{\theta}_n|\boldsymbol{\psi}_m, \mathbf{s}^t, \mathbf{x}^{t-1,M,N}, \mathbf{y}^{t-1,N}\right) \\
= & \eta p\left(\mathbf{y}_t^N|\boldsymbol{\theta}_n, \boldsymbol{\psi}_m, \mathbf{s}^t, \mathbf{x}^{t,M,N}, \mathbf{y}^{t-1,N}\right) \times \\
& p\left(\mathbf{x}_t^{M,N}|\boldsymbol{\theta}_n, \boldsymbol{\psi}_m, \mathbf{s}^t, \mathbf{x}^{t-1,M,N}, \mathbf{y}^{t-1,N}\right) \times \\
& p\left(\boldsymbol{\theta}_n|\boldsymbol{\psi}_m, \mathbf{s}^t, \mathbf{x}^{t-1,M,N}, \mathbf{y}^{t-1,N}\right) \\
= & \eta \underbrace{p\left(\mathbf{y}_t^N|\mathbf{s}_t, \boldsymbol{\theta}_n\right)}_{\sim N(\mathbf{y}_t;g(\mathbf{s}_t,\boldsymbol{\theta}_n),\mathbf{R}_t)} \underbrace{p\left(\mathbf{x}_t^{M,N}|\boldsymbol{\psi}_m, \boldsymbol{\theta}_n\right)}_{\sim N(\mathbf{x}_t;d(\boldsymbol{\psi}_m,\boldsymbol{\theta}_n),\mathbf{T}_t)} \times \\
& \underbrace{p\left(\boldsymbol{\theta}_n|\boldsymbol{\psi}_m, \mathbf{s}^{t-1}, \mathbf{x}^{t-1,M,N}, \mathbf{y}^{t-1,N}\right)}_{\sim N\left(\boldsymbol{\theta}_n;\boldsymbol{\mu}^{(k)}_{\boldsymbol{\theta},t-1},\boldsymbol{\Sigma}^{(k)}_{\boldsymbol{\theta},t-1}\right)}
\end{aligned} \tag{26}
$$

In above formula, $p\left(\mathbf{y}_t^N|\mathbf{s}_t, \boldsymbol{\theta}_n\right)$ and $p\left(\mathbf{x}_t^{M,N}|\boldsymbol{\psi}_m, \boldsymbol{\theta}_n\right)$ are robot observation to environment features and sensor network observation to environment features respectively.

In this solution, UKF algorithm is also used to estimate the environment feature location at present moment based on mean $\boldsymbol{\mu}^{(k)}_{\theta,t-1}$ and variance $\boldsymbol{\Sigma}^{(k)}_{\theta,t-1}$ estimated at (t-1)-th moment. The UKF processing is similar to sensor network calibration and not given because of limited paper space.

4 EXPERIMENTS AND ANALYSIS

Because the actual network robot system that integrates robot, sensor network and environment landmarks has not been built yet, the experiments are executed under a simulated system. Tim Bailey has provided simulations of SLAM and a dataset of 200mx200m [28], we develop the simulated system and set the landmarks and waypoints as shown in Fig. 2. The robot is located initially at the origin point orienting to left. The control period is set as $t_c = 0.025s$. Robot can obtain the range and bearing to the landmarks within front semicircle region of radius 30m, robot observation sampling period is set as $\Delta\mathrm{T} = 0.2$s and observation covariance is set as $\mathbf{R}_t = diag\{0.1^2, 0.1^2\}$, robot moves at rate $u = 3m/s$ with covariance $\mathbf{P}_t = diag\{0.3^2, 0.3^2, \left(3^\circ\right)^2\}$. The number of particle is set as $N = 100$. As shown in Fig. 2, 7 sensor nodes are fixed up to construct sensor network to monitor robot and the whole environment. Assume that each sensor can obtain range and bearing to both robot and landmarks within circle region of radius 30m, observation period is set as $\Delta\mathrm{T} = 0.05$s, and observation covariance of sensor nodes to landmark is set as $T_t = diag\{0.1^2, 0.1^2\}$, and to robot is set as $Q_t = diag\{0.1^2, \left(3^\circ\right)^2\}$.

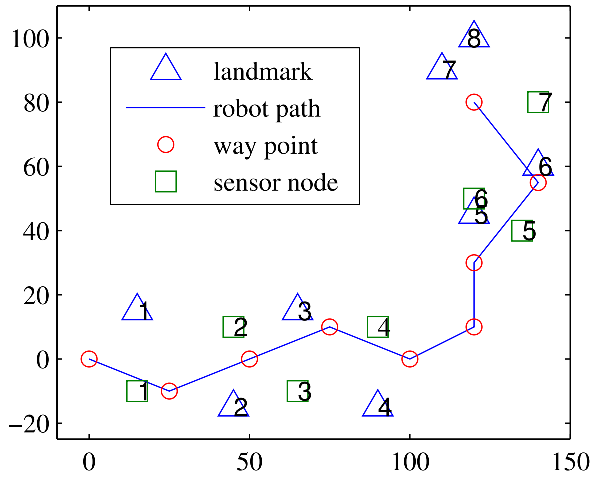

Fig. 2. Schematic diagram of environment features, sensor nodes and robot path

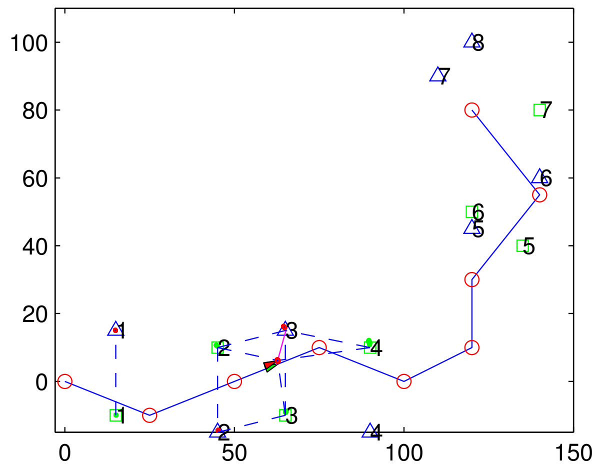

Fig. 3. Screenshot of simulation at 100ΔT

To intuitively show the estimated location of the robot, sensors and landmarks compared to their true location, Fig. 3 gives one screenshot of the simulation at 100ΔT, and at the same time Fig.4 gives an amplified screenshot of robot localization. The blue dashed lines denote observation of calibrated sensor nodes, and the red solid line denotes current observation of robot. Firstly, the robot localization accuracies are compared between traditional EKF-SLAM(without sensor network), FastSLAM 2.0(without sensor network) and SLCAM 1.0(network robot system simultaneous localization calibration and mapping with estimation technique of FastSLAM 2.0) and the proposed

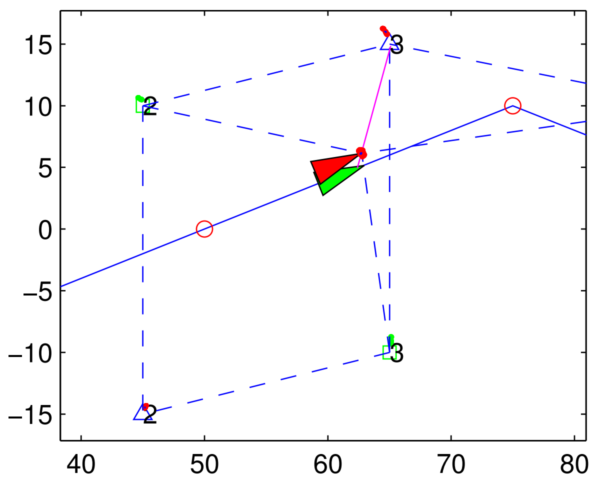

Fig. 4. Amplified screenshot of robot localization at 100ΔT

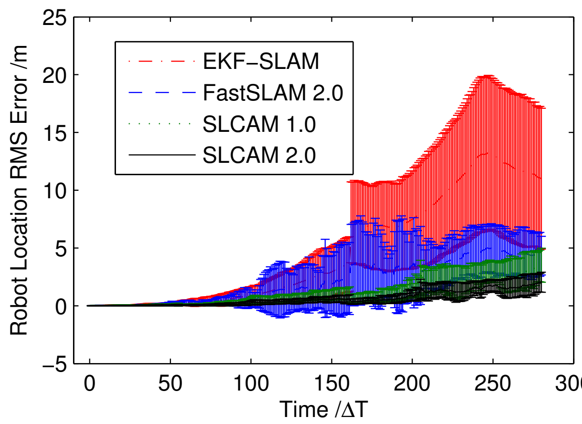

Fig. 5. Robot location RMS errors of 4 methods

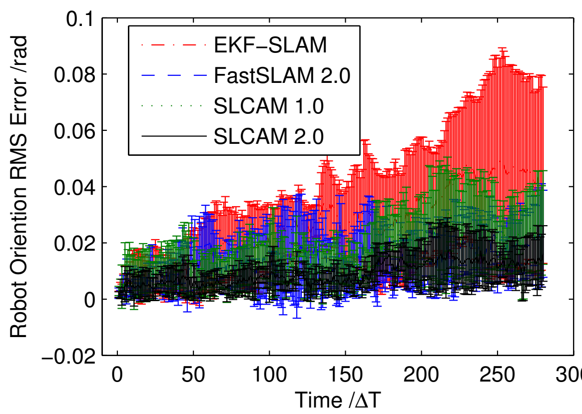

Fig. 6. Robot orientation RMS errors of 4 methods

SLCAM 2.0 (network robot system simultaneous localization calibration and mapping with EPF for localization and UKF for both sensor calibration and mapping). These algorithms run 50 times and the obtained means and variances of robot location root-mean-square (RMS) errors are given in Fig. 5, and means and variances of robot orientation RMS errors are given in Fig. 6. It can be seen that, the FastSLAM 2.0 full considers the robot observations to landmarks and improves localization accuracy obviously comparing to EKF-SLAM, but because robot sensor is fixed within robot, the localization errors are increasing faster with robot running both EKF-SLAM and FastSLAM 2.0. As for both SLCAM 1.0 and SLCAM 2.0, the department of sensor nodes from robot makes the sensor network observation and robot movement uncoupled, so both the accuracy and stability of the localization are improved. Furthermore, comparing to SLCAM 1.0, the SLCAM 2.0 obtains more accuracy and stability because it uses unscented transformation rather than approximating nonlinear models by neglecting higher order terms.

Secondly, the calibration accuracies are compared using only robot motion model, FastSLAM 2.0, SLCAM 1.0 and SLCAM 2.0. These algorithms run 50 times and the obtained means and variances of sensor network calibration RMS errors are given in Fig. 7. It can be seen that based on robot-motion-only method, the error of robot localization grows quickly along with robot motion distance, therefore the error of sensor network calibration is huge and hardly useful. FastSLAM 2.0 can deduce robot localization error and improve sensor calibration accuracies, but because it only considers the sensor observation to robot, the calibration accuracies of sensors are still not good enough. As for the SLCAM 1.0 and SLCAM 2.0, because of the independence between sensor network calibration and robot localization, the observation and motion are uncoupled, which makes the accuracies of robot localization and sensor calibration both high, but the stability of both estimation and calibration has improved. Besides, similar to deal with robot localization, SLCAM 2.0 performs better than SLCAM 1.0 in sensor network calibration. Thirdly, the mapping accuracies are compared using only robot motion, FastSLAM 2.0, SSEAM 1.0 and SSEAM 2.0. These algorithms run 50 times and the obtained means and variances of environment mapping RMS errors are given in Fig. 8. It can be seen that based on motion-only method, the error of robot localization grows quickly along with robot motion distance, therefore the error of environment mapping is also huge and hardly useful. FastSLAM 2.0 can reduce robot localization errors and improve mapping accuracies, but because only consider the observation from sensors to robot, the mapping error of sensors is still high. As for the SLCAM 1.0 and SLCAM 2.0, because of the uncouple relation between robot motion and sensor network observation, the accuracies of robot localization, sensor calibration and environment mapping are

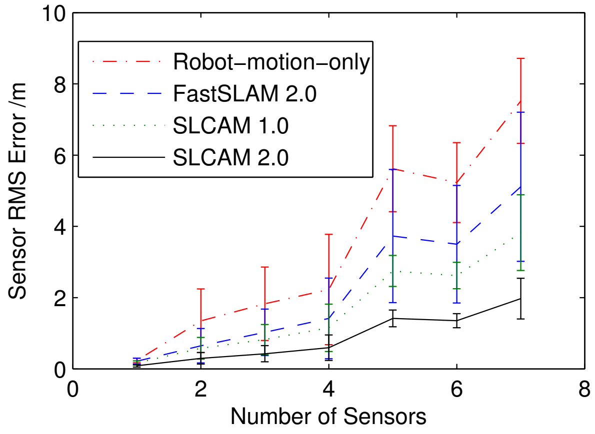

Fig. 7. Sensor location RMS errors of 4 methods

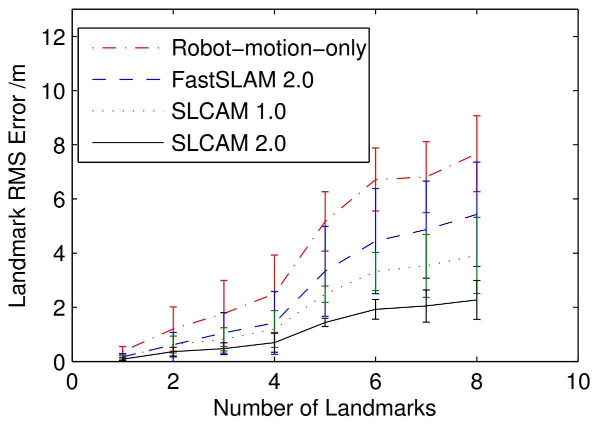

Fig. 8. Landmark location RMS errors of 4 methods

high and their stability is also improved. It can be also seen that SLCAM 2.0 performs better in environment mapping comparing to SLCAM 1.0. All the above experiments are executed using Matlab under Windows XP with CPU of Pentium 4 2.4GHz and 1G memory. The running times are respectively EKF-SLAM with 12.3s, FastSLAM 2.0 with 135.4s, SLCAM 1.0 with 580.8s and SLCAM 2.0 with 647.6s. It can be seen that although SLCAM 1.0 and SLCAM 2.0 consume more time than traditional SLAM, their efficiencies are still acceptable, and more importantly, the proposed methods can automatically calibrate sensor network once and for all after ubiquitous robot system is introduced into a service environment.

The experiments showed a relative balanced scatter of both sensor network nodes and environment landmarks, the proposed solution is also suitable for setup that contains more sensors and fewer landmarks, or vice versa, while the accuracy and stability may decrease.

5 CONCLUSION

Robot localization, sensor network calibration and environment mapping are three basic issues to make the network robot system works effectively. In this article, the problem of simultaneous localization, calibration and mapping in network robot system is put forward and denoted as joint conditional distribution. The problem is decomposed into three analytic terms in aspect of probability and a solution is given under the framework of Rao-Blackwellized particle filtering. Although the proposed methods are tested and verified under simulated system, to perform well under actual system, others issues such as wireless communication, feature detection, data association should also be considered, and the further optimization should be made so the algorithm could run in real-time.

As mentioned above, network robot system plays more and more important rule in daily life, especially in the dynamic environment. Based on solutions presented in this paper, the network robot system have calibrated the sensor network, and built a contemporary environment map. To ensure robot correct path planning and navigation, the map must be updated momentarily to reflect the change of environment. So how to update the map effectively through the interaction between robot and calibrated sensor network, comes to be an important research issue.

ACKNOWLEDGEMENT

This work is supported by State Key Laboratory of Robotics and System (HIT) in China No. SKLRS-2010-ZD-08, National Natural Science Foundation of China in China No. 61379065, No. 61305113 and Natural Science Foundation of Hebei province in China No. F2012203199.

REFERENCES

[1] S. Alberto, H. Norihiro, S. Alessandro, “Network robot systems”, *Robot. Auton. Syst.* Vol. 56, no. 10, pp. 793-797, 2008.

[2] J. H. Kim, I. B. Jeong, I. W. Park, et al, “Multi-layer architecture of ubiquitous robot system for integrated services”, *International Journal of Social Robotics*, vol. 1, no. 1, pp. 19-28, 2009.

[3] P. L.Wu, L. F. Kong, L. Kong, “A solution to the simultaneous robot localization and camera-network calibration problem of ubiquitous robot system”, *ICIC Express Letters*, vol. 5, no. 8, pp. 2759-2765, 2011.

[4] Y. B. Liu, J. H. Yang, Z. H. Wu, “Ubiquitous and cooperative network robot system within a service framework”, *Int. J. Hum. Robot*, vol. 8, no. 1, pp. 147-167, 2011.

[5] W. Yu, J. Y. Lee, Y. G. Ha, et al. “Design and implementation of a ubiquitous robotic space”, *IEEE T. Autom. Sci. placecountry-regionEng.*, vol. 6, no.4, pp. 633-640, 2009.

[6] C. Yu, X. Chen, "Home monitoring system based on indoor service robot and wireless sensor network". *Computers and Electrical Engineering*, vol. 39, no.4, pp. 1276-1287, 2013.

[7] H. L. Fu, H. C. Chen, P. Lin, "APS: Distributed air pollution sensing system on Wireless Sensor and Robot Networks". *Computer Communications*, vol. 35, no. 9, pp. 1141-1150, 2012.

[8] M. Luis, G. Andrew, C. Jesus, et al, "Data fusion in ubiquitous networked robot systems for urban services". *Annales des Telecommunications/Annals of Telecommunications*, vol. 67, no.(7-8), pp. 355-375, 2012.

[9] S. Thrun, "Simultaneous localization and mapping". *Adv. Robotics*, vol. 38, no. pp.13-41, 2008.

[10] H. Chen, K. Matsumoto, J. Ota, et al, "Self-calibration of environmental camera for mobile robot navigation". *Robot. Auton. Syst*., vol. 55, no.3, pp. 177-190, 2007.

[11] Z. W. Liang, X. D. Ma, F. Fang, et al, "Improved monte carlo localization algorithm in a hybrid robot and camera network". *Information Technology Journal*, vol. 9, no.8, pp. 1585-1597, 2010.

[12] F. Caballero, L. Merino, P. Gil, et al, "A probabilistic framework for entire WSN localization using a mobile robot". *Robot. Auton. Syst*., vol. 56, no. 10, pp. 798-806, 2008.

[13] D. Herrero, H. Martínez, "Range-only fuzzy Voronoi-enhanced localization of mobile robots in wireless sensor networks". *Robotica*, vol. 30, no. 7, pp. 1063-1077, 2012.

[14] G. Tuna, K. Gulez, V. C. Gungor, "Communication related design considerations of WSN-aided Multi-Robot SLAM", in *Proceedings book of IEEE International Conference on Mechatronics,* (Istanbul, Turkey), pp.493-498, 2011.

[15] D. Schleicher, L. M. Bergasa, M. Ocaña, et al., "Real-time hierarchical outdoor slam based on stereovision and GPS fusion". *IEEE T. Intell. Transp*., vol. 10, no. 3, pp. 440-452, 2009.

[16] E. Menegatti, A. Zanella, S. Zilli, et al., "Range-only SLAM with a mobile robot and a wireless sensor networks", in *Proceedings book of IEEE International Conference on Robotics and Automation,* (CityplaceKobe, country-regionJapan), pp.8-14, 2009.

[17] R. Wong, J. Z. Xiao, S. Joseph, "Data association for simultaneous localization and mapping in robotic wireless sensor networks", in *Proceedings book of IEEE/ASME International Conference on Advanced Intelligent Mechatronics,* (Montreal, QC, Canada), pp. 459-464, 2010.

[18] F. G. F. Angel, R. M. Mark, G. Jesus, "Multitarget simultaneous localization and mapping of a sensor network". *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4544-4558, 2011.

[19] Z. W. Liang, X. D. Ma, X. Z. Dai, "Simultaneous localization and mapping in a camera network". *Advanced Robotics*, vol. 24, no. 7, pp. 1037-1058, 2010.

[20] Y. Park, S. Jeong, I. H. Suh, et al., "Map-building and localization by three-dimensional local features for ubiquitous service robot". *Lect. Notes Comput. Sc*., vol. 4412, pp. 69-79, 2007.

[21] B. K. Kim, N. Tomokuni, K. Ohara, et al., "Ubiquitous localization and mapping for robots with ambient intelligence", in: *Proceedings book of IEEE International Conference on Intelligent Robots and Systems*, (Beijing, China), pp.4809-4814, 2006.

[22] J. Kelly, G. S. Sukhatme, "Visual-inertial simultaneous localization, mapping and sensor-to-sensor self-calibration", in *Proceedings book of IEEE International Symposium on Computational Intelligence in Robotics and Automation*, (placeCityDaejeon, country-regionKorea), pp. 360-368, 2009.

[23] Pham Viet-Cuong Juang, Jyh-Ching. A multi-robot, cooperative, and active SLAM algorithm for exploration. International Journal of Innovative Computing, Information and Control, 2013, 9(6): 2567-2583.

[24] Lee Heon-Cheol, Lee Seung-Hwan, Choi Myoung Hwan, Lee Beom-Hee. Probabilistic map merging for multi-robot RBPF-SLAM with unknown initial poses. Robotica, 2012, 30(2): 205-220.

[25] Aggarwal Priyanka, Syed Zainab, El-Sheimy Naser. Hybrid extended particle filter (HEPF) for integrated inertial navigation and global positioning systems. Measurement Science and Technology, 2009, 20(5): 1-9.

[26] R. Kandepu, B. Foss, L. Imsland, "Applying the unscented Kalman filter for nonlinear state estimation". *J. Process. Contr.,* vol. 18, no. (7-8), pp. 753-768, 2008.

[27] S. L. Lu, R. Zhang, "Two efficient implementation forms of unscented Kalman filter". *Control. Intell. Syst*., vol. 39, no. 1, pp. 1-11, 2011.

[28] T. Bailey, SLAM Simulations, http://www-personal.acfr.usyd.edu.au/tbailey/software/slam_simulations.htm.

P. Wu was born in Jinzhou (China) in 1981. He is currently associate professor at College of Information Science and Engineering, Yanshan University. He received his Ph. D. degree from Yanshan University in 2010. His research interest covers home ubiquitous robot system, Intelligent Space technology, SLAM.

L. Kong received his Ph. D. degree from Harbin Institute of Technology in 1995. He is currently full professor and Ph. D. supervisor of Computer Science and Technology at College of Information Science and Engineering, Yanshan University, where he is engaged in teaching and research. His primary interest covers home service robot, machine vision, intelligent information processing, parallel robotics and automatic control.

L. Kong received his master degree from Ocean University of China in 2007. He is lecturer at Hebei Normal University of Science and Technology. He is currently pursuing a Ph.D. in Computer Application Technology at Yanshan University. His research interest covers SLAM, target tracking.

S. Zhang received his PhD degree from Yanshan University in 2005. Currently he is a professor and supervisor for doctor's degree students at the College of Information Science and Engineering, Yanshan University. His research interests include visual information processing, pattern recognition and intelligent robot.

AUTHORS' ADDRESSES

Peiliang Wu,
E-mail: peiliangwu@gmail.com

Prof. Lingfu Kong,
E-mail: lingfukong@ysu.edu.cn

Liang Kong,
E-mail: kongliangouc@yahoo.com.cn

Prof. Shihui Zhang,
E-mail: sshhzz@ysu.edu.cn

School of Information Science and Engineering, Yanshan University, 438,Hebei Avenue, Qinhuangdao City,066004,Hebei Province, P.R.China.

Received: 2013-12-05
Accepted: 2015-01-27